SCIENCE • TECHNOLOGY • RESEARCH HIGHLIGHTS • VISIONS •

Dissertation

125

Lightweight adaptation to situational changes in classifiers of multimodal human data

Elena Vildjiounaite

VTT SCIENCE 125

Lightweight adaptation to situational changes in classifiers of multimodal human data

Elena Vildjiounaite

VTT Technical Research Centre of Finland Ltd

Thesis for the degree of Doctor of Technology to be presented with due permission for public examination and criticism in Oulu, at the University of Oulu, on May 20.

ISBN 978-951-38-8410-9 (Soft back ed.)
ISBN 978-951-38-8411-6 (URL: http://www.vttresearch.com/impact/publications)

VTT Science 125

ISSN-L 2242-119X
ISSN 2242-119X (Print)
ISSN 2242-1203 (Online)
http://urn.fi/URN:ISBN:978-951-38-8411-6

JULKAISIJA – UTGIVARE – PUBLISHER

Teknologian tutkimuskeskus VTT Oy
PL 1000 (Tekniikantie 4 A, Espoo)
02044 VTT
Puh. 020 722 111, faksi 020 722 7001

Teknologiska forskningscentralen VTT Ab
PB 1000 (Teknikvägen 4 A, Esbo)
FI-02044 VTT
Tfn +358 20 722 111, telefax +358 20 722 7001

VTT Technical Research Centre of Finland Ltd
P.O. Box 1000 (Tekniikantie 4 A, Espoo)
FI-02044 VTT, Finland
Tel. +358 20 722 111, fax +358 20 722 7001

Juvenes Print, Tampere 2016

Abstract

A wide range of current computer applications require explicit human-computer interaction of various types, ranging from application login to providing feedback on the applications' outputs (e.g., ranking recommended TV programmes), to adapt to different usage situations. As explicit interaction can be tiresome, users tend to avoid it even if such behaviour is counterproductive and/or insecure. Accordingly, application developers rarely build systems capable of runtime adaptation to new contexts, because conventional classifier training requires too large datasets of labelled training data to obtain from end users. The most common adaptation schemes define one or more typical usage contexts, build a pool of context-specific reasoning models during the design time and then select an appropriate model from this pool during the runtime. This approach enables instant runtime adaptation, but requires domain knowledge and suits only applications with usage contexts that can be pre-defined. Many personal applications, however, encounter wide varieties of difficult-to-define contexts, e.g. social rules or audio backgrounds. It is simply impossible to predict all such contexts, to say nothing of collecting adequate databases for building pools of reasoning models for them. Hence personal applications require new methods for adapting to changing runtime contexts. As runtime adaptation largely relies on interaction with end users, these methods should be fairly *lightweight* with respect to standard ones, i.e. they should require much less domain knowledge and explicitly acquired data.

This thesis introduces and explores lightweight solutions for building reasoning models for situations that are not pre-defined during the design time. These solutions are proposed for increasing the accuracy or convenience of applications in three domains: TV programme recommendations, affect recognition and personal assistance. In addition, a method for reducing explicit interaction efforts at the inference stage is proposed for increasing the security of biometric verification systems in which typical usage contexts can be pre-defined. The proposed methods have been validated experimentally with realistic data sets, and the results have confirmed that they considerably reduce the dependence of context- and user-adaptive classifiers on domain knowledge and explicit interaction efforts. Studies with personal assistive applications have also demonstrated that users can accept the proposed lightweight adaptation even when its accuracy is relatively low.

The diversity of test cases, which differed considerably in their requirements and data availability, made it possible to demonstrate how the suitability of different adaptation schemes depends on both the application and its usage contexts. Based on this experience, this thesis identifies context and application characteristics that exercise the greatest influence and provides guidelines for considering these factors in adaptation design.

Keywords: human-computer interaction, context adaptation, multimodal fusion

Tiivistelmä

Monet nykyisistä tietokonesovelluksista edellyttävät erilaista eksplisiittistä interaktiota ihmisen ja tietokoneen välillä. Tämä pitää sisällään esimerkiksi sovelluksiin kirjautumisen ja sovellusten toimintaa koskevan palautteen antamisen (esim. suositeltujen televisio-ohjelmien asettamisen paremmuusjärjestykseen) sovellusten mukauttamiseksi. Koska eksplisiittinen interaktio voi olla rasittavaa, käyttäjät mielellään välttävät sitä, vaikka tämä olisikin haitallista ja/tai tietoturvaa heikentävää. Sovellusten kehittäjät rakentavat sen vuoksi harvoin järjestelmiä, jotka pystyvät mukautumaan uusiin konteksteihin käytönaikaisesti, koska perinteinen luokittajan ohjattu opettaminen edellyttää liian suurien annotoitujen tietojoukkojen hankkimista loppukäyttäjiltä. Tavallisimmissa mukautumismalleissa määritetään yksi tai useampi tyypillinen käyttökonteksti ja rakennetaan suunnitteluaikana kontekstikohtaisten päättelymallien varanto, josta valitaan sopiva malli käytön aikana. Tämä lähestymistapa mahdollistaa välittömän käytönaikaisen mukautumisen, mutta edellyttää tietämystä toimialueista ja soveltuu ainoastaan sovelluksille, joissa käyttökontekstit on mahdollista määrittää etukäteen. Useita henkilökohtaisia sovelluksia käytettäessä vastaan tulee kuitenkin monia erilaisia, vaikeasti määritettäviä konteksteja, kuten sosiaalisia sääntöjä tai äänitaustoja. Kaikkia tällaisia konteksteja ei yksinkertaisesti ole mahdollista ennustaa, puhumattakaan niiden vaatimien päättelymallien varantoon riittävien tietokantojen keräämisestä. Tästä syystä henkilökohtaisia sovelluksia varten tarvitaan uusia menetelmiä, jotka mahdollistavat mukautumisen käytön aikana muuttuviin konteksteihin. Koska käytönaikainen mukautuminen pohjautuu pääasiallisesti sovellusten ja loppukäyttäjien interaktioon, näiden menetelmien on oltava melko kevyitä standardinmukaisiin menetelmiin nähden. Toisin sanoen niiden on edellytettävä vähemmän tietämystä toimialueista ja vähemmän eksplisiittisesti kerättyjä tietoja.

Tässä tutkielmassa esitellään ja tutkitaan kevyitä ratkaisuja päättelymallien rakentamiseen sellaisia tilanteita varten, joita ei ole määritetty ennalta suunnitteluaikana. Näitä ratkaisuja ehdotetaan sovellusten tarkkuuden tai kätevyyden parantamiseksi kolmella toimialueella: televisio-ohjelmia koskevat suositukset, tunteiden tunnistaminen ja henkilökohtaiset avustustoiminnot. Lisäksi ehdotetaan menetelmää, jonka avulla voidaan vähentää eksplisiittistä interaktiota päättelyvaiheessa suojauksen parantamiseksi biometrisissä todentamisjärjestelmissä, joissa tyypilliset käyttökontekstit ovat etukäteen määritettävissä. Ehdotetut menetelmät on vahvistettu kokeellisesti realististen aineistojen avulla. Saadut tulokset vahvistavat, että menetelmien avulla on pystytty tuntuvasti vähentämään kontekstin ja käyttäjän mukaan mukautuvien luokittajien riippuvuutta toimialuetietämyksestä ja eksplisiittisestä interaktiosta. Henkilökohtaisia avustavia sovelluksia koskevissa tutkimuksissa on myös osoitettu, että käyttäjät hyväksyvät ehdotetun kevyen mukautuksen, vaikka sen tarkkuus olisi suhteellisen heikko.

Koska testitapaukset olivat niin monimuotoisia ja poikkesivat huomattavasti toisistaan vaatimusten ja käytettävissä olevien tietojen osalta, oli mahdollista osoittaa, miten riippuvaista erilaisten mukautumismallien soveltuvuus on sekä itse sovelluksesta että sen käyttökonteksteista. Näiden kokemusten pohjalta tutkielmassa tunnistetaan kontekstien ja sovellusten ominaisuuksia, joilla on suurin vaikutus, sekä tarjotaan suosituksia siitä, miten nämä tekijät voidaan huomioida mukautumissuunnittelussa.

Avainsanat **human-computer interaction, context adaptation, multimodal fusion**

Preface

This thesis presents theoretical and experimental results from my work on four research projects in different application domains. The projects were carried out by different teams, but all the software needed for my research was developed by Vesa Kyllönen. Without his excellent skill, advice and sense of humour this thesis simply could not have appeared. My recent research visit to the University of Auckland, New Zealand, also contributed much to this thesis, and I am very grateful to Professor Georgy Gimel'farb (University of Auckland) for his constant support, encouragement and supervision. I am also grateful to my supervisor, Professor Tapio Seppänen (University of Oulu), for his invaluable advice and help in putting my results and publications together, and to Petteri Alahuhta for being a fantastic superior over a very, very long time. I also wish to thank Professor Oliver Amft for a thorough review of this thesis and very inspiring comments.

My research would have been impossible without the contributions and support of my superiors and colleagues. Heikki Ailisto convinced me that I should start thinking about a PhD, Julia Kantorovitch provided inspiring criticism, Satu-Marja Mäkelä and Johannes Peltola supported my interest in lightweight adaptation when this topic had just emerged, and Ville Könönen and Jani Mäntyjärvi helped me greatly during the fourth project and persuaded me to start writing the thesis. I am deeply indebted to Anna Sachinopoulou, Ilkka Niskänen, Johan Plomp, Jouni Kaartinen, Daniel Schreiber, Mikko Sallinen and many other colleagues for their cooperation and help.

Last, but not least, I appreciate the magnificent support I have had from my husband and our parents. I am also very grateful to the wonderful Finnish family Kilpelä, who have given us and our children warmth during the cold Northern winters, to our Russian friends who participated in several official and unofficial user studies belonging to my projects and provided invaluable feedback, and to Malcolm Hicks for the remarkable improvements he has made in the English of this thesis.

Elena Vildjiounaite, Oulu 2016

Academic dissertation

Supervisors Professor Tapio Seppänen
University of Oulu, Pentti Kaiteran katu 1, 90014 Oulu
Finland

Professor Georgy Gimel’farb
University of Auckland, Private Bag 92019, Auckland 1149
New Zealand

Reviewers Professor Oliver Amft
University of Passau, 94030 Passau
Germany

Professor Ari Visa
Tampere University of Technology, 33720 Tampere
Finland

Opponents Professor Jürgen Ziegler
University of Duisburg-Essen, 47057 Duisburg
Germany

Professor Ari Visa
Tampere University of Technology, 33720 Tampere
Finland

List of publications

This thesis is based on the following original publications, which are referred to in the text as Publications I–VI. They are reproduced with kind permission from the publishers.

I Vildjiounaite, E., Kyllönen, V., Ailisto, H., Empirical Evaluation of Combining Unobtrusiveness and Security Requirements in Multimodal Biometric Systems, *Image and Vision Computing*, Vol. 27 (2009) No: 3, 279–292. Author contribution: research problem, system design, multimodal fusion design and experiments for biometrics verification. Vesa Kyllönen: SW implementation. Heikki Ailisto: writing contribution.

II Vildjiounaite, E., Kyllönen, V., Hannula, T., Alahuhta, P., Unobtrusive dynamic modelling of TV programme preferences in a Finnish household, *Multimedia Systems*, Vol. 15 (2009) No: 3, 143–157. Author contribution: research problem, dynamic modelling design and experiments for TV recommender system. Tero Hannula: implementation of SW for retrieving TV programme metadata from the web. Vesa Kyllönen: implementation of all other software functionalities. Petteri Alahuhta: TV data acquisition.

III Vildjiounaite, E., Kyllönen, V., Mäkelä, S.-M., Vuorinen, O., Keränen, T., Peltola, J., Gimel'farb, G., Semi-supervised context adaptation: case study of audience excitement recognition, *Multimedia Systems*, Vol. 8 (2012) Issue 3, pp. 231–250. Author contribution: research problem, multimodal fusion design and experiments. Georgy Gimel'farb: supervision. Vesa Kyllönen: SW implementation for multimodal fusion. Others: context-independent audio and video analysis.

IV Vildjiounaite, E., Schreiber, D., Kyllönen, V., Ständer, M., Niskanen, I., Mäntyjärvi, J., Prediction of Interface Preferences with a Classifier Selection Approach, *Journal on Multimodal User Interfaces*, Vol. 7 (2013), Issue 4, 321–349. Author contribution: research problem, multimodal fusion design and experiments. Jani Mäntyjärvi: contribution to UI design and writing. Others: SW implementation, contribution to data collection and writing.

V Vildjiounaite, E., Kyllönen, V., Vuorinen, O., Mäkelä, S.-M., Keränen, T., Niiranen, M., Knuutinen, J., Peltola, J., Requirements and software framework for adaptive multimodal affect recognition, *Proceedings of the 2009 3rd International Conference on Affective Computing and Intelligent Interaction (ACII 2009). Amsterdam, The Netherlands, 10–12 Sept. 2009*, IEEE Press, 7 p. Author contribution: requirements, overall system design and experiments. Vesa Kyllönen: SW framework implementation. Others: context-independent audio and video analysis.

VI Vildjiounaite, E., Gimel'farb, G., Kyllönen, V., Peltola, J., Lightweight Adaptation of Classifiers to Users and Contexts: Trends of the Emerging Domain, *The Scientific World Journal,* 2015, Article 434826, 29 p. Author contribution: literature review and guidelines. Others: contribution to writing.

The following supplementary publications, closely related to the contents of the thesis but not included in it, can be separated out from the list of references:

S1. Vildjiounaite, E., Mäkelä, S.-M., Lindholm, M., Riihimäki, R., Kyllönen, V., Mäntyjärvi, J., Ailisto, H., Unobtrusive Multimodal Biometrics for Ensuring Privacy and Information Security with Personal Devices, In *Pervasive* (2006) 187–201.

S2. Vildjiounaite, E., Mäkelä, S.-M., Lindholm, M., Kyllönen, V., Ailisto, H., Increasing security of mobile devices by decreasing user effort in verification, In *Proceedings of the Second International Conference on Systems and Networks Communications* (ICSNC '07), IEEE Computer Society, Washington, DC, USA, 80–85.

S3. Vildjiounaite, E., Kyllönen, V., Hannula, T., Alahuhta, P., Unobtrusive Dynamic Modelling of TV Program Preferences in a Household, In *Proceedings of the 6th European Conference on Changing Television Environments* (EUROITV '08), Manfred Tscheligi, Marianna Obrist, and Artur Lugmayr (Eds.). Springer-Verlag, Berlin, Heidelberg, 82–91.

S4. Vildjiounaite, E., Kantorovitch, J., Kyllönen, V., Niskanen, I., Hillukkala, M., Virtanen, K., Vuorinen, O., Mäkelä, S.-M., Keränen, T., Peltola, J., Mäntyjärvi, J., Tokmakoff, A., Designing socially acceptable multimodal interaction in cooking assistants, In *Proceedings of the 16th International Conference on Intelligent User Interfaces* (IUI '11). ACM, New York, NY, USA, 415–418.

S5. Vildjiounaite, E., Kyllönen, V., Mäntyjärvi, J., If their car talks to them, shall a kitchen talk too? Cross-context mediation of interaction preferences, In *Proceedings of the 3rd ACM SIGCHI Symposium on Engineering Interactive Computing Systems (EICS '11).* ACM, New York, NY, USA, 111–116.

List of symbols

CBR	Case-Based Reasoning
CF	Collaborative Filtering
EER	Equal Error Rate
FAR	False Accept Rate
FRR	False Reject Rate
GUI	Graphical User Interface
ID	Identifier
HMM	Hidden Markov Model
kNN	k Nearest Neighbours
LSTM	Long Short-Term Memory Recurrent Neural Networks
MAP	Maximum a Posteriori
MLP	Multi-Layer Perceptron
MPM	Maximum Posterior Marginal
PSO	Particle Swarm Optimisation
SNR	Signal-to-Noise Ratio
SVM	Support Vector Machines
TV	Television
UI	User Interface

Contents

1. Introduction

1.1 Background

There are numerous applications nowadays that run on personal devices or devices in the environment and aim at providing support for their users in various tasks. This support often requires employing classification methods for different purposes, e.g. for user verification or for recommending the most appropriate items for the current user in the current situation. Situational changes may affect applications in two ways: (*i*) the convenience of an application may depend on the situation: e.g. the most appropriate GUI (graphical user interface) layout will depend on the screen size, and (*ii*) the application accuracy may depend on the situation: e.g. the accuracy of audio-visual analysis depends on environmental noise and lighting conditions.

Most often, adaptation to situational changes is performed by specifying typical usage situations at the design stage, building reasoning models for each of these situations and defining mappings between contexts and corresponding models. These mappings are then used for runtime model selection: e.g. different biometric fusion models can be selected for different predefined security levels [Kumar 2013], or different activity recognition models can be selected for different environments [Xu 2014]. While this approach may be the best option in application domains where recognition mistakes can entail high costs, the rapid increase in mobile technologies and databases for personal and group use is making the development of such situation-specific systems more and more problematic in many other domains, for three reasons.

First, the variety of usage situations may be so large, that it would be difficult to pre-define them all [Evers 2014], let alone collect adequate databases for developing and validating suitable reasoning methods for these situations. For example, although accounting for personal differences and social rules is an important goal for interactive systems [Aarts 2009, Dumas 2009, Dumas 2013, Evers 2014], affect recognition systems [Douglas-Cowie 2007, Vinciarelli 2012] and recommender systems [Borras 2014, Masthoff 2006], personal differences and social rules are culture-dependent and not easy to formulate. Model adaptation to new scenes is listed among the most significant challenges for intelligent environments [Shivappa 2010, Ye 2012], but the wide variety of activities that go on in these environments makes the concept of “scenes” somewhat elusive. Other examples of difficult-to-define contexts are personal goals (e.g., search intents) and variations between uncontrolled environments, e.g., mixed sounds. In such cases application designers often prefer to build non-adaptive systems. Thus it has been noted that the majority of human affect recognition approaches remain context-insensitive largely due to the difficulty of collecting and annotating contextual data [Zeng 2009, Vinciarelli 2012].

Secondly, even when pre-defining typical situations is feasible, this approach requires domain knowledge in order to define context classes and build reasoning models. Domain knowledge is generally acquired ad-hoc from an expert and applied to solutions that can hardly adapt to new conditions [Snidaro 2015]. For example, in multimedia analysis real-time event detection is often based on recognising small sets of context-specific sounds [Xu 2008, Otsuka 2009] and visual objects [Brezeale 2008, Hu 2011]: e.g. highlight detection may be based on sets of game-specific sounds particular to basketball and soccer matches [Xu 2008]. Due to the inflexibility of such systems, reducing the dependence on domain knowledge has been stated as one of the important challenges for multimedia analysis systems [Atrey 2010, Bhatt 2011]. Similarly, affect recognition systems may employ separate classification models for females and males [Kotti 2012, Tawari 2010] or for silent and talking users [Nguyen 2012], whereas recommender systems may employ different reasoning strategies for heterogeneous and homogeneous groups [Shin 2009, Sotelo 2009].

Last but not least, this approach allows applications to adapt only to coarse-grained context classes, not to peculiarities of each situation. Adaptation to specifics of each context can be achieved at the runtime, but existing methods for adapting to situations, emerging at the runtime stage, often rely on domain knowledge too. Usually, domain knowledge is employed in the form of various assumptions regarding user behaviour. A TV recommender system, for example, may employ the assumption that the preferences of family members who are older and earn money should dominate over those of other family members [Thyagaraju 2011]. This assumption does not hold good in all families, however, as parents may enjoy watching children's programmes together with their children, or one family member may dominate over the others irrespective of income or age. Other systems may employ other assumptions: e.g. that users who are similar to each other in one context will remain similar in others and that only the degree of similarity will change [Berkovsky 2008], or that certain types of users' interests will be valid in all contexts [Blanco-Fernandez 2010], or that recent contexts are more similar to each other than distant ones [Rafeh 2012]. Such assumptions do not always hold good, either.

Runtime adaptation would be more flexible if it were to be data-driven instead of relying on domain knowledge. But then the adaptation would be feasible only if the data collection efforts and the time required for them did not annoy the users. Recently, involvement of the end users into runtime adaptation process has gained importance [Evers 2014, Krupitzer 2015], but conventional fully supervised learning methods are scarcely suitable for this purpose because they require too large datasets for each context to acquire from end user [Schwenker 2014]. The majority of methods for adaptation to concept drift (i.e., changes in the relation between the input data and the system output (target variable) over time) also require human supervision, but a recent survey [Gama 2013] does not point out methods, suitable for learning from small datasets.

Conventional semi-supervised learning usually employs certain modelling assumptions, which may not hold in all contexts and may reduce the accuracy as compared with the use of labelled data only [Schwenker 2014]. Also, the difficulty of understanding human behaviour restricts the use of unsupervised learning because these approaches cannot adapt themselves quickly to peculiarities. Unsupervised data analysis may recognise that midday sleeping is not a typical behavioural pattern for the majority of adults, for example, but learning that it is a typical form of behaviour for a certain user would require either long-term observations of this person or human supervision. It is therefore important in the case of personal applications to develop efficient fully or partially supervised methods of runtime adaptation that use small sets of implicitly or explicitly acquired user data and do not expect noise-free data: user feedback may be inaccurate or erroneous.

1.2 Objectives

Personal applications, employing classification methods, may interact with users in two modes: training and inference. As users may be lazy about interacting explicitly even if such behaviour is insecure [Wright 2008], applications should not require too much explicit interaction in either mode. When context adaptation is performed by pre-defining typical situations, all training is performed during the system design time and thus explicit runtime interaction with the users is needed only for inference. When applications should adapt to contexts emerging during their run time, interaction with the users is also needed for classifier training. In this case classifiers should not rely on detailed domain knowledge, because it may not apply to new contexts.

Both cases require an understanding of how situational changes may influence user and system behaviour, and of which adaptation approaches are more appropriate for different application requirements. The pros and cons of implicit and explicit interaction depend on the applications and their usage contexts, as also does data availability. The feasibility of learning from scratch for each new context vs. reusing (transferring) knowledge from previously encountered contexts also depends on the application. Hence, the main objectives of the present work were two-fold:

- To suggest new and efficient methods of runtime adaptation for multimodal classifiers, in order to reduce the need for explicit interaction efforts and domain knowledge by comparison with conventional solutions;
- To identify the main characteristics of personal applications, strongly influencing adaptation design, and to provide guidelines on choosing between possible adaptation methods on the basis of these characteristics.

1.3 Scope

The present work is focused on the adaptation of multimodal classifiers that combine information from various sources via class- or decision-level fusion, because adaptation of feature-level fusion models typically requires much more computations. The classifier inputs, termed “cues” in the discussion below, can be time descriptors, interaction modalities, movie genres, audio classes or other data features, and the types of cues are assumed to be defined at the design stage. Whether an exact set of cues should be defined or not nevertheless depends on the reasoning algorithm. It is also assumed that the cues are provided by the same lower-level models in all situations, and that these will have been built at the design stage using an adequate development dataset. “Adequacy” means here that this dataset was collected for a context that was somewhat similar in its cue types to the contexts likely to be encountered during use of the application.

The work also adopts a common assumption of a “closed world” for classifier systems [Gama 2012], i.e. that the set of classes to be detected will remain the same and only the classification models will change (e.g. the same “interesting” vs. “uninteresting” output classes will hold good even when the recommender system is handling new items). Systems in “open worlds” should be able to recognise the limits of the current model and detect the emergence of previously unseen concepts, but existing methods for detection of new classes are highly sensitive to a threshold for the minimal amount of data samples required for consideration of emergence of a new class [Garcia-Borroto 2014]. Missing data present problems, too [Garcia-Borroto 2014]. Hence attempts to simultaneously learn output classes and corresponding classification models in new contexts, especially in ones where data availability is not guaranteed, are likely to require too large datasets and too long computations to be the first choice in adaptation design.

Context is generally defined as “conditions or circumstances which affect some thing” [Adomavicius 2011]. With respect to classification methods, the terms “situation” and “context” are often used interchangeably, so that “situation adaptation” and “context adaptation” both mean that some of the system inputs are treated differently from others in order to increase classification accuracy and/or user convenience. The term “context” often refers to external or latent factors which may influence user or system behaviour, e.g. knowledge regarding products and users’ buying histories is considered primary information in recommender systems, whereas the information that “Christmas is coming” is considered an external factor which may influence the users’ behaviour. More subtle distinctions between primary and contextual data are also fairly common: e.g. both audio processing results and dialogue acts (such as repetitions of user's statements) are derived from the same input data in spoken dialogue systems, but the dialogue acts are often referred to as context because they influence the interpretation of current user statements: the more often the user had to repeat his/her statements, the higher the probability of dissatisfaction even if the intonation used did not significantly change [Lopez-Cozar 2011]. Factors which affect data quality are also frequently called "context", e.g. the classification of audio data depends on the background (i.e. extraneous) noise. The term "context" may refer to fairly fine descriptors such as signal-to-noise ratio or video resolution, but it may also refer to higher-level abstractions such as events or locations.

We will speak here of adaptation to high-level entities as "contexts" or "situations", whereas their finer descriptors will be termed "context cues" or "context descriptors" (see Figure 1). For example, family habits are fairly high-level contexts, whereas time is a context cue. Situations may also overlap, for example, "weekend at home" context is a refinement of "at home" context.

Different context-adaptive systems employ different models of context. The most popular way of modelling context, called the "representational view" [Dourish 2004], is to describe it by means of a set of cues defined at the design stage, while another, far less common approach, called the "interactional view" [Dourish 2004], states that context cannot be described with a pre-defined set of cues, but rather the scope of context descriptors is defined dynamically in the course of human activities. Lightweight runtime adaptation to previously unseen contexts typically employs a mixed view, as illustrated in Figure 1, in which either exact sets or simply classes of context cues are defined at the design stage, whereas situations are defined dynamically on a runtime basis. Relevance of different context cues to the current user task can be determined at runtime too [Baltrunas 2014, Hussein 2014].

Situations can be defined dynamically by analysing primary data, but in computationally expensive ways, e.g. via unsupervised segmentation of a set containing mixed data applying to several contexts [Yu 2009, Tang 2012], or by training several classifiers on different data chunks and comparing their outputs [Yasumura 2007]. It is assumed here that the context change was detected in a different way, either by dedicated sensors or via user interaction. In the latter case the users may declare such a context change by explicitly naming a new context, or may indicate it implicitly by correcting classification errors and requesting adaptation. The former case requires employing context recognition algorithms. Development of such algorithms is beyond the scope of this work.

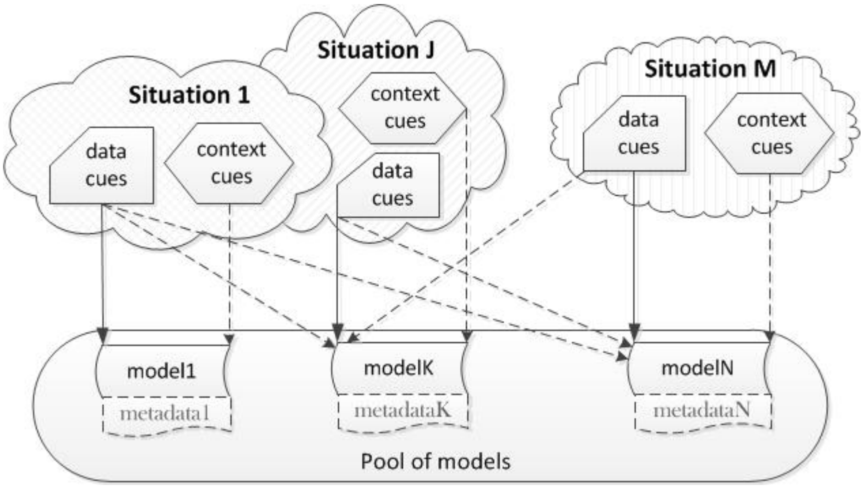


Figure 1: Modelling situation dependence. Dashed lines denote optional data.

Situational adaptation problem is related to concept drift adaptation problem. Concept drift is a change in the relation between the input data and the system output (target variable) over time; it can be categorised as real (changes in the conditional distribution of the output given the input data) and virtual (changes in distribution of the input data) [Gama 2013]. Situational changes often affect both distributions; hence, situational adaptation often requires adapting to virtual and real concept drifts simultaneously. Concept drift can be also categorised as abrupt, gradual, incremental or reoccurring, and knowledge regarding the type of drift helps to manage reasoning models, e.g., to decide whether to discard or store old ones [Gama 2013]. Similarly, situations may change incrementally or gradually and/or overlap, or may change abruptly and be non-overlapping. Naturally, situations may also re-occur. In this work we focus on development of methods, suitable for adapting to various types of situational changes, and suggest that model management should depend on the adaptation method and context recognition method.

The present work focuses on adapting multimodal fusion models on the assumption that the same lower-level models can be used in all contexts. This approach fails in contexts where low-level models fail, but it has certain advantages, too. First, as shown by [Guz 2010], it is more efficient to update the later stages of cascaded systems rather than the earlier ones. Secondly, it is easier for users to provide explicit feedback on the final classification result. Moreover, this feedback is more reliable, because users do not see the outputs of the lower-level models as frequently as they do the final classification results, and they may be confused by a need to review the lower-level results. Third, there is no need to propagate the user feedback to the lower levels as this may be unreliable if different modalities contribute to the fusion result in different ways depending on the context.

Runtime context adaptation of an application is referred to here as ***lightweight*** if its costs are considerably lower than those of conventional training of the same application and are therefore acceptable to the end user. The costs include data collection, annotation, reasoning-related computations, communication costs and opportunity costs (the latter are losses, occurring when the system does not deliver classification results because of model updates) [Zliobaite 2015]. This notion is necessarily informal, because the concept of “user acceptance” can scarcely be quantified, for it depends on the user’s personality, the perceived benefits of using the application, the convenience of the user-application interaction, and many other non-quantitative psychological factors. Reduction in costs of data collection and computations depends on the application too: some works, reviewed in Publication VI, reported that users had to spend only a few minutes for data labelling, while conventional approach would have required an hour or more. In some other cases computational time was reduced from few hours to a few minutes.

In a typical conventional learning or adaptation scenario all the reasoning models required by the application, e.g. single modality classifiers and multimodal fusion models, are built or modified, respectively, by intricate supervised or semi-supervised learning techniques. The latter reduce the need for explicit interaction efforts at the cost of employing substantial computational resources and domain-specific modelling assumptions. Lightweight adaptation reduces the explicit interaction efforts in various ways, most of which need less computational resources and modelling assumptions, but call for specific kinds of classifiers. In particular, lightweight adaptation may be based on unobtrusive data acquisition, and/or employ cascaded classifiers, and/or split the development of a customised application into two successive stages. In a cascaded architecture the lower levels are fully trained at the design stage using context-independent data and only the topmost reasoning models are learned or modified during the runtime to adapt to a context. In a two-stage development scenario the models are trained first for one or more typical contexts at the design stage on the basis of a large annotated database and then adapted to a new context during the runtime.

To the best of our knowledge, research into adaptation has not yet provided any guidance on choosing justified trade-offs between adaptation costs and the accuracy achieved. The proposals for adapting algorithm granularity (finer or coarser data clustering) to computational resources and users' needs [Gaber 2006, Haghighi 2009], for example, did not guide the choice of adaptation parameters. Works, suggested that system evaluation should include assessment of efforts, required for achieving system goals [Khaleghi 2013], did not provide generic guidelines either, while works, proposed to consider model adaptation as investment decision and to assess its return on investment, argued that adaptation utility depends on a particular source of data, i.e., on application [Zliobaite 2015]. Furthermore, the perception of how difficult data labelling is depends on the person concerned [Settles 2009], and user satisfaction with the accuracy of a particular classification output is also dependent on personal and other factors, including screen size: it has been shown that the predicting of UI (user interface) preferences with as low an accuracy as 50% was already beneficial for users of small devices, whereas users with larger screens needed higher accuracy [Findlater 2008].

To account successfully for a rich variety of personal preferences, it is suggested here that one should choose the adaptation granularity in a practically effective way, by employing lightweight methods whenever they are feasible and performing a finer adaptation either at the user's request, or during the application's idle times.

1.4 Author's contributions

This work focuses on lightweight situational adaptation of multimodal classifiers under different application requirements and on methods for rendering these classifiers user- and context-specific without any significant explicit interaction efforts or use of domain knowledge, especially assumptions regarding influence of context on user and/ or system behaviour. The main contributions of the author were as follows:

1. Design of multimodal fusion modules and adaptation solutions (Publications I–V);
2. The development of novel methods for using implicit interaction data and unlabelled data for reducing explicit interaction efforts and building situation-adaptive classifiers in various applications, in particular:
    a. Methods for cascaded inference by using implicit interaction data as long as its accuracy satisfies the application requirements, and requesting explicit interaction efforts to complement implicit data otherwise (Publication I);
    b. Methods for employing classifier ensembles, in which base classifiers use context descriptors as input cues, and diversity criteria for choosing base classifiers in context-adaptive ensembles (Publication II);
    c. Methods for fast statistical learning on small databases of noisy user-labelled data which can be used either for employing unlabelled target context data in cascaded training, or for transferring knowledge from previously encountered contexts (Publication III);
    d. Methods for building classifier ensembles, in which different members model different approaches to the transfer of knowledge from previous situations to a new one and data for different contexts are used for learning which transfer model is best suited for each context transition (Publication IV);
3. Design of a software framework for context adaptation (Publication V) and demonstration of the applicability of the suggested lightweight adaptation methods to various practical problems by presenting experimental results obtained in four application domains differing significantly in their requirements, data availability and degree of similarity between contexts (Publications I–IV);
4. Analysis of the influence of various context types on classification methods in several application domains, identifying important application characteristics and presenting recommendations on how these factors can be considered in lightweight situational adaptation (Publication VI).

The four application domains chosen here cover a fairly large proportion of all human activities: biometric verification is used for accessing various kinds of personal and work-related data, for moving inside office buildings and during travelling; TV recommender systems help people to relax; adaptation of the interfaces of assistive applications facilitates safety and enjoyment during the performing of various personal tasks; and affect recognition can be used for finding the highlights of TV programmes or in memory aids. In two of these domains, TV programme recommendations and affect recognition, the adaptation was aimed at increasing the classification accuracy, whereas in the other two domains (biometric verification and interface adaptation) the aim was to increase user convenience. Each test case provided unique experiences, on account of the notable differences in application requirements, such as the permitted adaptation time, data availability and the variability of the usage contexts (i.e. whether an application will be used in just a few contexts or in many, and whether these contexts are easy to define). Only implicit interaction data were used in the TV recommender test case, whereas explicit interaction data and unlabelled data were used in the emotion recognition case and a mixture of implicit and explicit interaction data in the other two test cases.

The requirements in terms of adaptation time and the ability to handle context variations in the four test cases (biometrics, recommender systems, emotion recognition and interaction adaptation) are illustrated in Figure 2, and the differences between these cases from the viewpoint of the approaches to adaptation and data usage adopted here are shown in Figure 3.

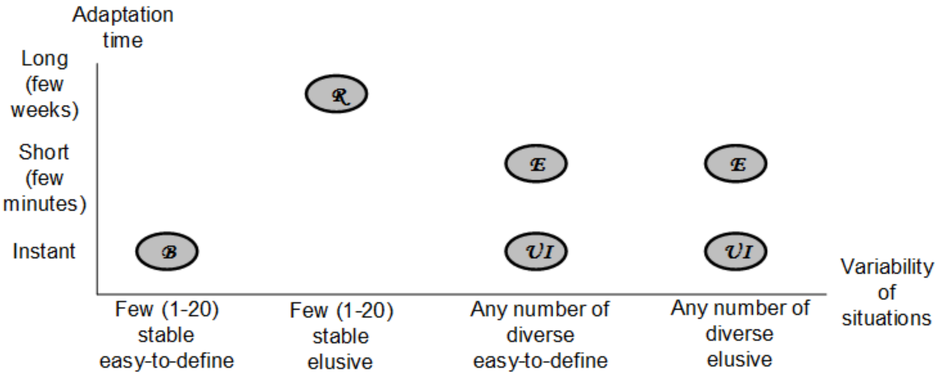


Figure 2: Requirements in terms of adaptation time and the ability to handle context variations in the four test cases: *B* – biometrics; *R* – recommender systems; *E* – emotion recognition; *UI* – interaction adaptation.

In the biometric verification case adaptation was performed by pre-defining a set of situations and training all the classification models at the design stage, on account of the security requirements, while in the other test cases methods for the runtime learning of reasoning models for new situations were proposed. A method of instant adaptation to both easy-to-define and elusive contexts was suggested in the UI adaptation test case, a method of quick adaptation to any new context (taking just a few minutes) was suggested in the emotion recognition test case, and methods for long-term learning were studied in TV recommender case. In all the cases the most important design choices were associated with the adaptation approach and the use of data applying to multiple contexts. Thus only target context information was used in the recommender systems test case because this was available in sufficient quantities and because users preferred this way. In the other cases methods were developed for reusing data from other contexts for adaptation purposes, as only very small quantities of target context data were available. The choice of single classifiers vs. classifier ensembles depended on data availability and the permitted adaptation times. These choices will be described in more detail in Sections 2 and 3.

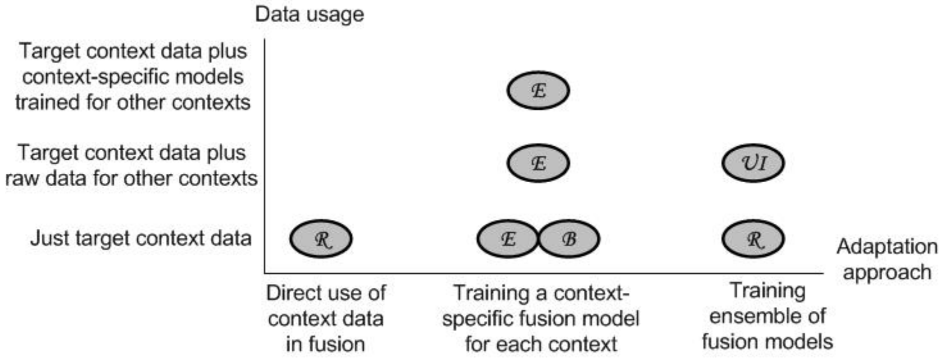

Data usage
Target context data plus context-specific models trained for other contexts
Target context data plus raw data for other contexts
Just target context data
E
E
UI
R
E
B
R
Adaptation approach
Direct use of context data in fusion
Training a context-specific fusion model for each context
Training ensemble of fusion models

Figure 3: Choices of adaptation type and data usage in the four test cases: *B* – biometrics; *R* – recommender systems; *E* – emotion recognition; *UI* – interaction adaptation.

2. Review of the literature

The requirements for lightweight adaptation differ significantly between application domains. Adaptation in biometrics has to satisfy security requirements, whereas adaptation mistakes in other domains are not likely to cause serious problems for the users. Accordingly, adaptation to limited numbers of pre-defined situations predominates in biometrics, while methods for adapting to large varieties of contexts have to be considered in other domains. Most often the classifiers are adapted to historical, social, task-specific, environmental and computational contexts.

- Historical factors embrace anything in the past that may affect the current state, e.g. user or system actions (recently viewed movies or verification attempts), changes in the user's mood or appearance over time, etc.
- Social factors include rules and customs affecting human interaction, e.g. gender/age-dependent behaviour; what is considered polite in different situations, etc.
- Task-related factors represent the objectives of specific users, e.g. the purpose of an information search, time available, etc.
- Environmental factors are anything in the surroundings that may affect sensor readings, e.g. background noise, light etc.
- Computational factors specify system settings, e.g. availability or quality of a certain data type (such as image resolution), computational power, algorithm capabilities, etc.

Lightweight adaptation is an emerging area of research, and not many approaches have been proposed to date. Accordingly, although this work is mainly concerned with the influence of context on user behaviour and preferences, relevant methods for adapting to differences in users' personalities will also be reviewed here. Affect recognition systems, for example, need to consider the fact that people usually express emotions less freely in formal settings than in informal ones. Reserved people, however, will always express their emotions more subtly than more open ones. In this case adaptation to personal differences does not differ conceptually from adaptation to "formal vs. informal settings" in the sphere of context.

Nevertheless, certain popular approaches to the reduction of user effort are not reviewed here, e.g. general-purpose unsupervised and semi-supervised learning methods, because they are either based on inflexible modelling assumptions or the computations involved are too intensive. Semi-supervised learning is often based on the assumption that points located close to each other in a feature space belong to the same class [Schwenker 2014, Zhou 2010]. It is well known, however, that points that are close to each other in one context may appear to be quite distant in another context. Consequently, the modification of a similarity measure is a quite common method for adapting to users and contexts [Luan 2011, Anand 2010]. Unsupervised data analysis favours statistically significant patterns and may thus fail to learn atypical context-dependent forms of behaviour. Active learning methods are not reviewed here, either, as these methods choose the data samples that are most informative for machine learning algorithms, but these are not necessarily easy for humans to annotate. Moreover, the perception of labelling difficulty depends on the individual [Settles 2009], so that it is easier for users to choose for themselves which samples to annotate.

2.1 Major design decisions

Adaptation can be seen as a generic machine learning problem: how to build a system so that it will perform well enough on the data it encounters during its run time. The main differences between conventional and context-adaptive approaches are two-fold:

1) Conventional approaches assume that runtime data do not differ dramatically from training data, whereas situational adaptation may require the system to take account of cases of significant differences between contexts;
2) Conventional classifiers typically aim at high recognition accuracy, whereas lightweight context-adaptive classifiers aim at being user-friendly even at the cost of somewhat decreased recognition accuracy.

Nevertheless, the major design decisions in the two types of system are fairly similar. Designing a conventional system requires us first to choose whether to use a single classifier or an ensemble of multiple classifiers. In the latter case, additional decisions must be taken on the combination, classifier, feature and data levels [Kuncheva 2004]:

1) On the combination level it must be decided how to deal with the outputs from the "base classifiers" (members of the ensemble), and in particular how to select the members for each case, and/or how to combine their outputs.
2) On the classifier level it is a matter of which base classifiers to choose, e.g. whether to employ the same algorithm or different ones.
3) On the feature level it must be determined whether all the base classifiers should use the same input features or different ones, and which features should be chosen.
4) On the data level it is a question of whether the same or different data sets should be used to train all the base classifiers, and, if necessary, how to choose the training data in order to select the best classifiers and/or optimise their combination.

Based on the choices between single and multiple classifier systems and then on the design choices on the combination and data levels, Publication VI classifies context-adaptive systems as follows:

- *Context as a Feature* (single and multiple classifier systems using context cues as latent factors, nodes in graphical models, or input features):
    - *Mixed Data Models:* models trained on data for the target and other contexts – usually single classifier systems;
    - *Context-Specific Models:* models, trained on target context data – usually used in multiple classifier systems;
- *Model selection* (a separate model is trained for each context and either discarded after context changes or stored along with metadata describing the context to which it is applicable – for later retrieval by the metadata):
    - *Context-Specific Classifiers and Data*: a model for each context is trained on the target context data and context-specific features and/or reasoning algorithms are employed;
    - *Context-Specific Models and Data*: a model for each context is trained on the target context data, but feature selection and the reasoning methods are the same for all contexts;

    - *Knowledge Transfer*: feature selection and the reasoning methods are the same for all contexts, and knowledge of other contexts (in the form of reasoning models, raw data, etc.) is used to build models for the target context;
- *Ensemble* (several models are trained, and either their outputs are combined or the best model is selected based on certain criteria):
    - *Context-Specific Ensemble*: each base classifier and/or the selection/combination methods are trained on the target context data;
    - *Mixed Data Ensemble*: the base classifiers use both the initial and target context data and their selection/combination methods are optimised on the target context data.

The choices between the above options are the most important ones because they determine system behaviour. A multiple classifier system, where different classifiers are optimised for different contexts, can switch arbitrarily and abruptly between contexts, whereas a single classifier system using context parameters as features would react more smoothly to context changes. Due to the possibly significant differences between contexts, it is also important to decide whether only data for the target context should be used for adaptation to the target context or whether data or models for other contexts should be used too. Knowledge transfer, or transfer learning, has emerged recently as a new learning framework to address the need to reduce data labelling efforts, but the decision whether to transfer knowledge or not is often based on distance between the initial and target domains, estimated according to some similarity measure [Pan 2010]. Defining similarity measures usually requires domain knowledge and hence it is difficult to design similarity measures, suitable for everybody. For example, a businessman may perceive context “Friday evening” as notably different from context “Monday morning”, whereas a retired person may perceive the difference as insignificant. Hence user-friendly situational adaptation requires methods to transfer knowledge between both similar and dissimilar contexts without explicitly estimating similarities between contexts.

The choices at the classifier and feature levels depend on the details of the problem in question, as in conventional systems, but feature selection and algorithm selection are usually performed once for all contexts because tailoring these to each context would require additional training data.

The proposed classification comprises classifications of adaptation approaches, suggested in [Macias-Escriva 2013, Snidaro 2015]. Macias-Escriva et al. [Macias-Escriva 2013] considered only number of inference models and stated that adaptation in a multi-model system is a procedure of switching between models, whereas adaptation in a mono-model system is a procedure of tuning its parameters. Snidaro et al. [Snidaro 2015] considered only use of contextual data: use of context as constraints (forbidden operations, probabilistic conditioning etc.) vs. use of context as additional features, semantics or situation elements (e.g., context may change a meaning of information or bring new dimensionality into a problem). The work [Macias-Escriva 2013] does not list adaptation via selection of multiple models and combining their results, though, and the work [Snidaro 2015] is concerned with use of context features rather than high-level situations.

2.2 Adaptation in the biometric domain

2.2.1 Research topics and open issues

The majority of the reported work on biometric verification is focused on a multi-biometric recognition setup based on a fixed set of information sources [Deravi 2012]. Thus if a biometric system provides access control for applications with different security requirements, it will be tuned to satisfy the highest security requirements. Hence False Reject Rates (FRRs) for accessing low security applications would be unfeasibly high, as well as user efforts. Accordingly, increased user convenience is said to be one of the main goals for the next generation of biometric systems [Jain 2012, Deravi 2012, Zhang 2014], this being achievable by the following means:

- Dynamic selection of biometric modalities in order to satisfy the requirements of a particular task (typically the security level) and user preferences (e.g. dislike of a sensor for hygienic reasons or incapability to provide the required samples due to a trauma);
- Employing unobtrusive biometric modalities in instant and/or continuous verification. The latter is usually performed periodically, but it was also suggested to increase security level by means of verifying the user after every action, e.g., every mouse action [Mondal 2015]

The main open issues in this domain are:

- Adaptation to task factors (the various security requirements and choices of biometric modalities);
- Adaptation to the reliabilities of biometric modalities (mainly influenced by computational factors and user characteristics).

The reliabilities of modalities may also depend on environmental factors in uncontrolled settings, but this issue is rarely studied, because interaction with biometric sensors usually takes place in controlled settings. The recent rapid increase in the sale of smartphones and tablets initiated research into continuous unobtrusive verification in uncontrolled conditions, but this is still only an emerging research topic. Adaptation to computational factors, such as cross-device adaptation, is also an emerging research area. Such adaptation can help to reduce explicit user efforts by allowing users to enrol on a laptop and to be verified by a mobile phone (and vice versa), as for example in [Khoury 2014]. Syed et al. [Syed 2014] suggested to employ knowledge transfer for adapting to computational factors (e.g., touch dynamics and application usage dynamics depend on operating system of a mobile device, while web browsing patterns may be platform-independent), but we are not aware of any studies into this problem.

Social factors influence biometric systems mainly indirectly: e.g. a society's attitude towards certain biometrics may affect personal preferences. Historical factors do not play an important role, either, because the verification and identification results typically depend only on system inputs obtained within a very short time interval. It has been suggested recently that biometric systems should be adapted to variations in biometric data caused by long-term system use (e.g. changes in user appearance), but research into this problem is mainly concerned with updating templates of individual modalities in unsupervised ways [Roli 2008, Poh 2009], as the supervision of end users may facilitate spoofing. To the best of our knowledge, incremental updates of biometric fusion models have not yet been studied, and this problem is not listed among the main challenges for the next generation of biometric systems [Jain 2012].

Due to security requirements, training data for biometric systems are usually acquired and manually labelled under the supervision of service providers.

2.2.2 Model selection

One of the simplest approaches for adaptation to the reliabilities of biometric modalities is to estimate them from training data, with the False Accept Rate (FAR) or Equal Error Rate (EER) serving as a reliability indicator, and employ them directly in the Weighted Sum fusion method, as in the work of [Nageshkumar 2009], for example.

Alternatively, weights can be estimated by means of minimising error rate of multimodal fusion with training data [Sim 2014]. Similarly, weights of a linear logistic regression fusion method can be calculated using a development dataset [Khoury 2014]. In [Mukherjee 2014] fusion of scores employed a polynomial instead of weighted sum. All exponents and coefficients were estimated using differential evolution algorithm, aiming at minimising overlap between distributions of genuine and imposter fusion results.

Adaptation to various security requirements is usually achieved by pre-defining a set of security levels and training a fusion model for each level so that the system error (often FAR or EER) does not exceed the desired value. [Kumar 2010], for example, suggested employing four score-level fusion methods (Weighted Sum, Weighted Product, Weighted Exponential Sum and Weighted Tan-Hyperbolic Sum) and searching for optimal weights and thresholds for these methods with the particle swarm optimisation algorithm (PSO). Although this approach provides for continuous degrees of security, the authors suggested employing 20 discrete security levels instead, because a parameter search in the case of continuous degrees should be performed at the moment of verification and this would cause the system response to be too slow. The same four score-level fusion methods and 20 security levels were employed by [Kumar 2013], but the parameter search was performed using the ant colony optimisation algorithm instead of the PSO. Giot and Rosenberger [Giot 2012] proposed estimating the parameters of various fusion methods (Weighted Sum, Weighted Product, Minimum of Scores, Maximum of Scores, etc.) via a genetic programming algorithm with the EER serving as a fitness function.

Works on adaptation to the availability of biometric modalities follow two approaches. The first is to train models for different combinations of modalities and to select a model for fusion on the basis of the list of modalities it can handle [Fatukasi 2008, Vera-Rodriguez 2012]. Another approach is to treat unavailable modalities as missing data and to apply generic methods to fuse all modalities in presence of missing data, such as imputation of the missing samples or modification of the fusion algorithm [Aste 2015]. Suitability of generic methods, however, depends on so-called “missingness mechanism”, i.e., whether data are missing at random or not [Aste 2015]. For example, imputation methods do not fit well the “data are missing not at random” scenario [Aste 2015], whereas “data are missing at random” assumption is infeasible when context influences availability of modalities (e.g., if etiquette requires silence, voice data will be unavailable). Hence several methods for handling missing data were developed specially for biometric systems. [Nandakumar 2009] proposed a new rank combination statistic for identification systems that was based on Bayes’ decision theory. This method was found to be more beneficial for the task of retrieving the top few matches than for finding the best match, however. [Wang J 2011] proposed modifications to SVM (Support Vector Machines) -based verification systems that would allow them to handle missing data. Comparison of the two approaches demonstrated that handling missing data via model selection achieves higher accuracy than using a single fusion method [Fatukasi 2008, Wang J 2011].

Our approach to the case study of biometric verification in Publication I therefore employed model selection. We aimed at increasing users’ convenience by using unobtrusive modalities and by allowing users to select modalities dynamically, whereas only a few other works have attempted to satisfy both requirements simultaneously, and only one of these allowed users to choose the modalities. Takahashi et al. [Takahashi 2004] employed cascaded fusion to deal with variations in users’ choices, so that when a user provided a biometric sample, the system either accepted this or allowed the choice of another modality. Decision-level fusion of modalities was performed using a Sequential Probability Ratio Test. The ability of the proposed method to keep FAR within the desired limits was proved in experiments on a database of five people, but unfortunately no results indicative of overall system performance (the FRR that can be achieved with different system configurations) were presented.

Another approach to reducing user effort and at the same time maintaining the desired security level was suggested by Allano et al. [Allano 2010], but this did not allow the users to choose the biometric modalities. Instead, the order of the modalities was fixed at the design stage based on their accuracies, obtained on the training data, so that the users had first to provide biometric samples of the most accurate modality, then of the second best, etc. Fusion was also performed using the Sequential Probability Ratio Test. Similarly, Erzin et al. [Erzin 2005] selected the order of modalities based on their reliability, but estimated the reliabilities dynamically, by calculating the difference between the best and second best likelihood ratios. This estimation method is based on the assumption that the correct model would create a significantly higher likelihood ratio than any other model. These approaches reduce overall user effort because they allow the skipping of some modalities, but users' preferences are ignored.

Adaptation to using sample quality in fusion is usually performed in one of the following ways [Poh 2012]:

- Cluster-based: quality measures are first grouped into a number of clusters and then a fusion model is built for each cluster;
- Feature-based: using quality measures directly as features.

Poh and Kittler [Poh 2012] presented a general Bayesian framework for using quality measures in fusion and compared the accuracies of the cluster-based and the feature-based approaches when used for several tasks. The model selection approach achieved higher accuracy in the tests, especially when the number of quality measures was increased. Feature-based approaches will be reviewed in the section on "Using context as a feature". The cluster-based approach has been successfully employed by Poh et al. [Poh 2007] and Fatukasi et al. [Fatukasi 2007]. The former work used just two quality clusters for facial images: good and poor illumination, while the latter work used three for faces (good illumination, left illumination and right illumination) and two for speech (signal-to-noise ratio (SNR) and "entropy quality").

As the use of quality measures as input features increases the complexity of the model, more training data are required for learning such models. Thus Fierrez-Aguilar et al. [Fierrez-Aguilar 2004] instead used score qualities for penalising SVM training errors, assigning lower penalties to misclassifications of lower-quality samples. This approach enables the multimodal system to focus on the modalities of the dominant qualities, improving fusion robustness, but as a result the system is optimised for good quality data instead of being able to handle different quality levels. Similarly, Casale et al. [Casale 2012] optimised their accelerometer-based gait recognition system to verify users only when they were walking. In this system user activity recognition results were used instead of sample quality: gait data of all non-walking activities, such as running or climbing stairs, were ignored. This approach increased accuracy of unobtrusive verification, but nevertheless it would not suffice to satisfy high security requirements.

These studies suggest that it is best to use non-discriminative context parameters for clustering/splitting primary data, and the present work followed this approach.

2.2.3 Ensembles

Ensembles have been proposed in biometrics only for adaptation to a historical context. The performance of the multimodal system may deteriorate if the distributions of the scores for individual modalities change due to changes in their templates, if these are compromised due to security breaches. Biometric systems usually store data in encrypted form and can thus recover by cancelling the compromised templates and replacing them with new ones created using another transformation algorithm. [Canuto 2013] suggested employing a stacked ensemble of multimodal fusion classifiers for preventing system degradation in such cases, the ways of combining the outputs from the ensemble members being fixed at the design stage. The feasibility of this approach was confirmed via experiments with two modalities and three transformation algorithms.

Connolly et al. [Connolly 2013] used ensembles for adaptation to changes in faces. They employed an incrementally growing classifier pool, selecting the optimal ensembles from this pool by means of the particle swarm optimisation algorithm, but this required over 3.500 training samples.

2.2.4 Context as a feature

Most often sample quality is used as an additional input feature in the fusion algorithm, for example, the quality of audio/visual scores estimated from SNR, audio signal entropy, face angle and illumination served as input features to SVM, GMM and logistic regression classifiers [Kittler 2007]. The use of score quality in fusion does not necessarily increase accuracy, however, as was demonstrated in the tests with SVM, MLP and Bayes classifiers and confidence estimates derived from score distributions [Bengio 2002].

The problem of preserving the desired security level while using unobtrusive modalities was studied in [Sim 2007], where face and fingerprint data were acquired using a video camera and a computer mouse with an embedded fingerprint sensor, and time interval since the last verification served as a context. Fusion was performed by the Hidden Markov Model (HMM) with two hidden states (“safe” and “attacked”) and biometric scores as observations. Adaptation to different security requirements was achieved by comparing the probability of the “safe” state with a predefined threshold, where a higher threshold enabled higher security. This method suits well for unobtrusive verification because it can easily handle differences in the reliabilities of modalities and missing data, but purely unobtrusive verification may be insufficient for high security applications, and it is unclear whether such a system can switch quickly between security levels.

2.3 Adaptation in the multimedia analysis and retrieval domain

2.3.1 Research topics and open issues

In this domain classifiers are mainly employed for event or concept detection purposes. The main open issue in multimedia analysis is adaptation to difficult-to-define computational contexts, i.e. to differences between multimedia types (e.g., TV genres) and sources (e.g., databases and TV channels), which usually incorporates the influence of environmental factors, too. This is an important problem because the accuracies of algorithms trained on one multimedia type or source and tested on another are usually 1.5–2 times lower than within-type and within-source accuracies [Yang 2009, Yao 2012]. One of the reasons for such behaviour is the use of domain knowledge, e.g. for selecting the audio/visual cues specific to each genre. Reducing the dependence on domain knowledge is therefore one of the serious challenges for multimedia analysis [Atrey 2010, Bhatt 2011].

On the other hand, systems aiming at overcoming dependence on domain knowledge by employing generic features usually recognise only fairly high-level concepts. For example, Shyu et al. [Shyu 2008 employed only generic features (e.g. dominant colour ratio, background variance, audio volume and spectrum etc.) in a decision tree classifier for concept detection in two rather different datasets: sports videos and news broadcasts on two TV channels. The classifier was trained to recognise five concepts, whereas the use of context-specific features typically allows the detection of tens or hundreds of concepts.

Social factors do not influence multimedia analysis significantly because fine-grained human behaviour understanding is rarely needed. Environmental, computational and task-related factors may require handling missing data and synchronising data from different modalities, but these issues can typically be dealt with fairly straightforwardly, by accumulating data within a pre-defined time window [Atrey 2010, Bhatt 2011]. Historical factors are important only for detecting long-lasting events.

The main open issue in multimedia retrieval is adaptation to difficult-to-define task contexts, notably user queries. This is an important problem on account of significant differences between users and their goals (i.e. social and task-related factors). Another open issue is the need to learn from very limited user feedback (just a few labelled samples) [Thomee 2012] and to do it quickly. Hence multimedia retrieval systems are often built up hierarchically, so that the lower layers perform context and user-independent multimedia analysis and upper layers are iteratively adapted by re-ranking the results from the lower ones [Thomee 2012].

Another open issue is learning from imbalanced and/or noisy data. As manual annotation is tiring, labelling of multiple categories may result in over 10% erroneous labels [Joshi 2012]. Thus it is more common to ask users to correct system results than to provide labels from scratch. It is also common to allow users to select for themselves which items to annotate [Kirstein 2012], but this often results in imbalanced data. The quality of implicitly obtained labels depends on the interface design, but they are in any case not perfect. For example, it has been suggested that retrieval results should be displayed as small image thumbnails and enlarged when the user clicks on them [Cheng 2009], but this approach does not provide feedback on images that are easy to understand without clicking. Clicked results are not always relevant to the user’s search either [Ghorab 2013]. This problem is usually dealt with by employing generic noise-tolerant classifiers, e.g. SVM [Thomee 2012, Zhang 2009].

2.3.2 Model selection

One way to reduce dependence on domain knowledge is automatic detection of data cues that are characteristic of each case. The content-adaptive framework developed by Radhakrishnan et al. [Radhakrishnan 2006] for audio analysis in domains such as sports and surveillance videos, where interesting events do not occur frequently, detected such characteristic sounds as outliers with respect to the usual events. Lu [Lu 2009] detected characteristic sounds in various video genres by reference to the frequency of their occurrence, by analogy with the popular “term frequency – inverse document frequency” weighting scheme used for text retrieval. Such unsupervised approaches usually require less data labelling effort than supervised ones, but a human supervisor is still needed for attributing the resulting characteristic sounds to the appropriate semantic descriptors, e.g. for specifying which sounds show the audience’s interest.

A significant drawback in these approaches is the need for computational power at the time when the adaptation results are to be produced. Thus real-time classification systems usually employ pre-trained models, but in such systems both feature selection and classification models are usually tailored to selected contexts [Bhatt 2011, Rehman 2014]. For example, based on results of [Radhakrishnan 2006], Otsuka et al. [Otsuka 2009] developed a personal video recorder that detected highlights in sports videos via the recognition of a small number of sound classes specific to sports contexts, the most important being a mixture of excited commentator’s speech and spectators’ cheering. Xu and Chua [Xu 2006] employed different sets of visual features and different algorithms (rules vs. SVM vs. HMM) for detecting events in soccer and American football. Xu et al. [Xu 2008] detected highlights in three types of sports by recognising two or three sounds specific to each (e.g., whistling and bouncing of the ball in basketball games vs. long whistles, double whistles and multiple whistles in soccer) in addition to four more generic sounds occurring in all the types (excited and plain commentator speech, excited and plain audience sounds). Even works employing fairly generic video or audio classes often rely on domain knowledge to a certain extent. Silence, hitting of the ball and applause were selected as audio classes for highlight detection in racket games, for example [Zhu 2007], and the same classes could be useful for detecting certain events in other contexts, too (e.g. in circus performances) provided the detection of audience excitement did not rely on estimation of the duration of the applause, since duration is a good indicator of excitement only in contexts where the audience can applaud for as long as it wishes, but in many other contexts, including circuses, the applause stops as soon as the artists start talking.

In our approach [Publication III] the same generic cues were used in all contexts. We did not study whether classification accuracy would notably increase if we were using context-specific cues instead, but some later works have demonstrated that a good initial choice of features may help to avoid extensive feature selection for each new context. Kirstein et al. [Kirstein 2012] compared the use of the same features vs. feature selection in the fully supervised learning vector quantisation method that they employed for object categorisation. In their tests with fairly challenging images containing objects of different shapes and colours rotated at various angles, the use of the same features for different object categories did not notably reduce the accuracy as compared with feature selection.

When the same cues are employed in all contexts a fairly simple form of adaptation can be achieved by context-dependent weighting of the cues [Atrey 2010]. The most common way of doing this is to pre-define sets of situations and employ a kind of “if – then – else” strategy, so that a video modality is given a higher weight than an audio modality during the daytime, for example, but the opposite is the case at night. Weights can be estimated based on the performances of various modalities with training data or by reference to prior knowledge [Atrey 2010, Shivappa 2010]. Ways of adapting to changes in the runtime reliabilities of modalities without relying on domain knowledge and training data have also been suggested, such as estimating the dispersion of modality outputs, e.g. by comparing several of the topmost scores or by estimating stream entropy [Shivappa 2010].

A fairly common approach for adapting multimedia retrieval systems is to modify the feature weights in a similarity measure (“feature relevance estimation” approaches) or in a query vector (“query vector modification” techniques) [Thomee 2012]. The easiest way of combining inputs is a weighted sum, but other schemes have also been suggested, e.g. products and square roots [Calumby 2012]. Weights can be adapted using heuristics or genetic algorithms: the latter do not require domain knowledge but do require training data. We have tested context-dependent weighting in the biometric and UI domains [Publications S1, S2, S5] and have found it inferior to more sophisticated methods.

A good alternative to re-weighting is to tune a classifier to handle small dataset sizes by selecting appropriate classifier parameters, data features or training samples and training it on the target context data only. These approaches often employ SVM due to its good generalisation capabilities [Ferecatu 2008, Luan 2011]. Probabilistic approaches such as Bayesian inference have also been tested in multimedia retrieval systems, but evidently require more feedback data than re-weighting-based approaches [Yin 2005].

Methods for avoiding the complete re-training of classifiers have been studied in research areas “transfer learning” and “domain adaptation”. In such cases the model for the target context is built using also knowledge regarding the initial context (although the terms “domain” and “task” are typically used in these areas rather than “context”). Related research areas are multi-task leaning and meta-learning: the former aims at the simultaneous learning of several classification problems, usually on the assumption that some model parameters can be shared between tasks [Evgeniou 2004], and the latter aims at exploiting knowledge regarding the performance of various classification algorithms with a variety of datasets for predicting which algorithm and/or which set of algorithm parameters will be most likely to learn new data successfully [Lemke 2015]. Unfortunately, the majority of existing works on meta-learning assume that the selected algorithm is then trained on the new data from scratch [Lemke 2015]. Furthermore, meta-features are often data characteristics that cannot be reliably computed from small datasets, e.g. correlations between input features [Lemke 2015].

Domain adaptation and transfer learning may take place on the data level, feature/representation level, parameter level, relational level or model level (also called the “function level”) [Yang 2009, Pan 2010]. The majority of the proposed methods are fairly computationally expensive [Morvant 2012, Patel 2015, Pan 2010], especially the ones on the data level.

Domain adaptation on the data level aims at finding a projection of the source data onto the target data so that their distributions will become closer. This approach does not require re-training the classification model [Morvant 2012]. Transfer learning on the data level aims at finding a way of using some of the labelled data from the source domain for training a model for the target domain [Pan 2010]. Often higher penalties are assigned to classification errors in the target data than to errors in the source data. This approach aims at increasing the amount of training data in the target domain, but it does not eliminate the need to fully re-train the classifier. In addition, this approach requires the storing of raw data for all contexts and may fail in cases where the current context differs significantly from others.

Transfer learning on the parameter, relational and model levels can be more lightweight. On the parameter level it can be based on a domain-dependent choice of model parameters to be shared across several tasks. It is fairly common, for example, to assume that the priors of Gaussian Process models are shared, but not other parameters [Pan 2010]. This approach reduces the need for training data, because fewer parameters have to be estimated for each context [Yang 2009, Pan 2010]. Parameters can be also optimised by means of an evolutionary algorithm, as in the work [Pauplin 2010], for example, where users evaluated the quality of segmentation, i.e. indicated whether the images were segmented into too many or too few regions, and the segmentation parameters were modified accordingly. Similarly, only selected model parameters were optimised in our model-level knowledge transfer method for Hidden Markov Models (HMM) used in an affect recognition application [Publication III].

The main idea of relational-level knowledge transfer is to learn analogies between entities and their co-occurrences. Learning analogies between entities has been proposed for relational domains such as ontological knowledge representations, social networks etc. The knowledge that a professor is the superior of a student, for example, or a movie director is the superior of an actor, can help to build mappings between other entities in the learning and filming domains, respectively [Pan 2010].

Learning relations between output classes in different contexts has been employed for increasing the accuracy of concept detection in multimedia analysis. Concept detection is a multi-class classification problem in which the classes are not exclusive, and adaptation can be performed by learning which classes co-occur with each other most frequently. The various approaches to learning such relations fall into two major groups [Qi 2008]. In the first, hierarchical modelling is used to train binary classifiers for each concept separately and then an additional classifier, SVM or logistic regression, for example, is employed to fuse their results. Alternatively, instead of a classifier, semantic graphs of relations between concepts can be built up on top of concept detection models [Jiang 2012], or association rules can be employed [Bharadwaj 2014]. In the second approach a model of the relations between concepts is constructed directly from low-level features, often also in the form of a graph [Weng 2008, Weng 2012] or a random field [Qi 2008]. Comparison of the two approaches [Qi 2008 and Jiang 2012] shows that they can achieve similar levels of accuracy, but the hierarchical approach is significantly faster. The work [Qi 2008] showed the proposed non-hierarchical method to be 25 times slower than the hierarchical approach, while in the work of Jiang et al. [Jiang 2012] adaptation using the proposed hierarchical approach required tens or hundreds of seconds (depending on the size of the dataset), whereas the training of non-hierarchical models would require tens or hundreds of hours. We employed hierarchical architectures in all our test cases.

Transfer learning on the model level is performed by modifying the parameters of a model for an old context using labelled data for the target context. Yang et al. [Yang 2007, Yang 2009, Yang 2012] employed SVM for concept detection in videos and proposed to represent a target classifier as the sum of a source classifier $f_s(x)$ and a delta function $\Delta f(x)$, defining a boundary shift:

$$f(x) = f_s(x) + \Delta f(x).$$

This delta function is learned from the training data via minimisation of the regularised empirical risk, so that both the classification error and the distance between the target and source classifiers are minimised. Yang et al. [Yang 2007, Yang 2009] studied two important issues that are rarely addressed in other works because of the effort required. First, they compared the accuracies of three approaches to the building of context-specific SVM models with the accuracy of an ensemble of SVM models trained separately on data for different contexts and combined using the sum of weights denoting the importance of the new context). The comparison involved the following context-specific models:

- model-level knowledge transfer;
- training on the merged data for the old and new contexts;
- training on the data for the target context only.

In the tests on data for 13 TV channels the average accuracy of model-level knowledge transfer was very close to that of the models trained on the merged data for different contexts, but the training times were 13–15 times shorter. Training on the data for the target context only resulted in lower accuracy due to the small number of positive training samples (ranging from one to ten). The accuracy of the ensemble was fairly close to that of the lightweight adaptation, but sensitive to weight choice. Yang [Yang 2009] additionally evaluated a semi-supervised SVM model trained on labelled and unlabelled target context data, but without using knowledge regarding the initial contexts. This semi-supervised SVM appeared to be significantly less accurate than the model-level knowledge transfer. Secondly, Yang et al. [Yang 2007, Yang 2009] studied a problem of selecting from among the existing context-specific models the ones that were most suitable for adaptation to the target context, but none of the approaches significantly outperformed the others in the tests.

The problem of selecting the most suitable initial model(s) can be eliminated by adapting a so-called general model instead, i.e. a model trained on data for all the initial contexts, but this approach requires the storing of raw data. Several studies have confirmed the feasibility of adapting general models for HMM using Gaussian mixture models. Adaptation of the maximum a posteriori (MAP) classifier was applied in [Zhang 2005] to the detection of meeting activities (such as note-taking, discussion, etc.), and event-specific models were obtained from a fairly small amount of labelled data. Modification of mixture parameters from a general model to a speaker-adapted one via linear regression with the minimum classification error was used successfully in [Wu 2007], so that adaptation using only four to six minutes of speech data achieved a significant increase in recognition accuracy for continuous speech.

As allowing only a small shift in the SVM decision boundary [Yang 2007, Yang 2009] may hinder adaptation in cases of significant differences between the initial and target contexts, Jiang et al. [Jiang 2008] suggested another SVM adaptation method, to train a model on a dataset containing labelled data for the target domain and support vectors from the initial model, where the support vectors are weighted according to their distances from the new training samples (longer distances are penalised). This method requires additional data for choosing the weights, since if they are too small the old knowledge will be practically ignored, whereas if they are too high they will not allow proper adaptation. Li et al. [Li 2012] proposed one more SVM adaptation method that involved training a model on a dataset containing labelled samples from both the source and target domains but choosing the training samples for labelling by a combination of two strategies: 1) taking samples from close to the decision boundary of the initial model, and 2) taking the samples that are most unlikely to be generated by the initial domain data distribution (modelled via the kernel density estimation approach). The former strategy is better suited to cases of similar initial and target domains and the latter to cases of dissimilar ones. In order to avoid similarity estimation, the strategies were combined as follows: the more samples suggested by a certain strategy at the current iteration receive positive labels, the larger the proportion of samples that this strategy will choose at the next iteration. In experiments on detecting 36 concepts in two fairly different video collections the proposed approach outperformed those suggested by Yang et al. [Yang 2007, Yang 2009] and Jiang et al. [Jiang 2008], but adaptation was not very fast, as it required ten iterations.

Due to above-described drawbacks of semi-supervised training and active learning, our work did not employ these methods. To provide for significant differences between contexts, in the affect recognition test case fairly big changes of model parameters were allowed [Publication III].

2.3.3 Ensembles

A fairly typical way of building ensembles is to train the base classifiers on different datasets, as it increases their diversity. Training different base classifiers on data of different contexts in combination-based ensembles may fail in cases involving fairly dissimilar contexts, however. For example, in a study into sensor-based activity recognition [Casale 2015] two combination-based ensembles were compared: in the first ensemble base classifiers were trained on data of different contexts, and in the second ensemble base classifiers were trained on the data of the target context only. In some target contexts accuracies of both ensembles were fairly close to each other, but in one target context the accuracy of the first ensemble was by 30% lower than the accuracy of the second one. Hence Patricia and Caputo [Patricia 2014] proposed more sophisticated approach: to use outputs of classifiers, trained on different source domains, as input features of a target domain classifier. In the tests involving concept detection task in images this approach outperformed single classifiers, trained on target context data only.

A more common approach is to train diversity-based ensembles on target context data only. Zhang and Ye [Zhang 2009] proposed training different base classifiers on the target context data using all the labelled positive examples and negative examples, randomly selected from unlabelled data (so that the datasets contained different negative examples). This ensemble was first used for removing wrongly labelled samples, employing a technique in which each ensemble member had to classify all the positive examples and the samples classified as negative by all the ensemble members were considered wrongly labelled. A new ensemble was then trained on the reduced training dataset and the final result was obtained by combining the outputs of all the ensemble members. This approach did not require long training times in the tests because just five positive training samples were labelled for each query. Our recommender system [Publication II] employed a similar principle, in which the classifier ensemble included SVM and CBR (case-based reasoning) classifiers, and the SVM was trained using both positive and negative examples while the CBR used only positive examples. The diversity of the base classifiers in our ensemble was not due only to the use of different training data, however (for more details, see Section 3.3.4).

A combination-based ensemble may also employ context-independent base classifiers. Shih et al. [Shih 2009], for example, employed an ensemble of generic attention models for retrieving sports videos in which each base classifier estimated the viewer's attention based on a certain modality, such as camera motion, object detection, etc. The adaptation was lightweight, as the weights of the models were adapted on the basis of feedback from each user.

Adaptation via the selection of the best ensemble members is also lightweight. Dynamic selection strategies choose the most appropriate reasoning method(s) for each test sample based on its similarity to training samples [Britto 2014]. This similarity estimation is usually based on input features, as in the work of Mporas et al. [Mporas 2011], where adaptation to different noise conditions was achieved by choosing between several speech enhancement algorithms. Cavalin et al. [Cavalin 2012], on the other hand, proposed computing output scores for all the ensemble members with respect to the test sample and comparing these scores with the scores for the training samples, because similarities between the input features were not preserved across contexts. The ensemble was trained incrementally by adding new classifiers to a pool and removing the least frequently used ones. Learning how to select the best subset for each sample nevertheless required increasing the training dataset by 20–25% compared with the data needed for training the base classifiers.

Another way of selecting the most appropriate reasoning method(s) from a pool is to evaluate the performance of the methods with the target context data. Yin et al. [Yin 2005] employed an ensemble of relevance feedback strategies and reinforcement learning for selecting the most appropriate strategy for each query and image class. In addition, they compared the use of target context data only (i.e. interaction data for a current query) with the use of mixed data (i.e. interaction data for multiple retrieval sessions performed by multiple users). The use of mixed data in the tests led to a greatly increased precision in the initial results, those images retrieved before using the relevance feedback. The precision after the first iteration using the relevance feedback also increased, but less notably, whereas the improvements after the second iteration were insignificant. Our selection of best classifiers [Publications II, IV] was organised by comparing their performance when used on the target context data.

Ensembles in which the base classifiers are re-trained during the runtime can be employed for dealing with concept drift, as this can be detected by comparing the predictions obtained with the classifier trained on the latest data chunk, with those of classifiers trained on the old data. For inference, either the predictions of only the latest classifier can be used, or else the outputs from the old classifiers can be re-weighted, or all the classifiers can be re-trained on the updated dataset [Yasumura 2007].

2.3.4 Context as a feature

In multimedia analysis the term “context” most often denotes internal image or video characteristics, such as spatial relations between different image regions [Katuka 2014] or temporal relations between video segments [Papadopoulos 2011]. Contextual cues can be also taken from various external sources, e.g. location data or social networks data. Even demographic statistics can serve as context: in [Gallagher 2008] it was used in a graphical model for estimating a probability that a person of a certain age and gender has a certain name. The main goal in such cases is to better exploit additional data for each situation independently.

2.4 Adaptation in the recommender systems domain

2.4.1 Research topics and acquisition of training data

The main difference between recommender systems and information retrieval systems is that the former always take long-term personal preferences into account, whereas the latter always focus on short-term user interests related to the user’s current task and expressed in the form of a query. Although recent information retrieval systems employ long-term interaction histories too, adaptation to a current query remains their main goal. On the other hand, the main goal of recommender systems is to adapt to the different personalities of users, which are assumed to be fairly stable, whereas adaptation to current interests is still an emerging trend, in that earlier recommender systems did not model the dependence of users’ preferences on the task (e.g. finding a gift for a friend vs. a gift for a spouse) and nor did they consider social factors (e.g. whether a person is watching a movie alone or with children). Therefore the use of context in recommender systems is generally an open issue, so that one fairly recent survey on context-aware recommender systems states that developing a better understanding of how to use the context in recommender systems is an important but largely unsolved problem [Adomavicius 2011]. The two main sub-problems here are:

- adaptation to social context;
- adaptation to computational and task-related factors.

In both cases adaptation to a target context can be enhanced by using preferences, acquired for another context (the "mediation of preferences" [Berkovsky 2008]). For example, preferences of an individual can be used for adaptation to social context; preferences for books can be used in a movie recommender system or movie ratings, stored in one movie recommender system, can be used in another movie recommender system.

Both types of context are difficult to define. Adaptation to social context is important because there are many social activities in this domain, e.g. watching TV and travelling. Although some systems handle social context by pre-defining a few coarse classes, e.g., "alone" vs. "in company" [Adomavicius 2005, Baltrunas 2012], other works suggest that adaptation to social context is a challenging problem that deserves special treatment [Jameson 2007, Masthoff 2006, Senot 2010, Garcia 2014]. The mediation of preferences is important because recommender systems often employ explicit user interaction for acquiring ranks of items (e.g. movies) or item attributes (e.g. movie genres, actors etc.). The context for which these preference values are acquired is also often obtained explicitly or via sensors. Therefore the mediation of preferences would allow a reduction in the explicit ranking effort and would also help to handle the "cold-start" problem (the problem of providing recommendations for a new user or a known user in a new context). The majority of current recommender systems provide recommendations only for users who have already expressed their preferences in contexts that match the target context either exactly or in some generalised form [Adomavicius 2011]. The mediation of preferences would also allow a reduction in data collection time in systems acquiring user preferences implicitly, by observing how users deal with recommended items. For example, fairly common indicators of implicit satisfaction with TV programmes, movies and music are 1) selecting vs. skipping items in the recommendation lists, and 2) the percentage of the item duration viewed/ listened to by the users [Holbling 2010, Stober 2013]. Such percentages have indeed been shown to correlate with user satisfaction [De Moor 2011].

Adaptation to other context types is usually carried out by pre-defining their classes. For example, adaptation to task-related and environmental contexts is often performed by pre-defining a few classes of budget, weather, etc. [Baltrunas 2012], while adaptation to historical factors other than concept drift is usually performed using application-dependent heuristics. A user who has just bought an expensive computer, for example, is not likely to buy another the next day, whereas a user who just watched a comedy may watch another comedy immediately afterwards.

2.4.2 Model selection

Recommender systems often employ collaborative filtering (CF), a reasoning method based on the assumption that users who have behaved in a similar fashion in the past will to some extent continue to do so in the future. CF reduces the need for separate data for each user by using data on user communities, and thus our UI adaptation solution [Publication IV] employed CF. CF-based context adaptation approaches can be classified as pre-filtering, post-filtering or contextual modelling [Adomavicius 2011]. In the latter case the context is often used as a feature. Pre-filtering and post-filtering employ recommendation methods developed for context-independent systems, with pre-filtering using only target context data, whereas post-filtering first provides recommendations using data acquired in all contexts and then either filters or re-ranks the results using contextual information. Music recommendations, for example, can be provided using all data and then re-ranked using the historical context, e.g. the last songs just listened to by the user [Hariri 2012]. Panniello et al. [Panniello 2009] compared the use of exact pre-filtering and two post-filtering approaches (filter vs. re-rank) in CF, their main conclusion being that only the best post-filtering method can outperform pre-filtering, but finding the best post-filtering method for each task may require expensive search protocols. Further comparison of the pre-filtering and post-filtering techniques did not result in a clear winner with respect to different evaluation metrics either [Panniello 2014].

Recently, research has turned towards employing CF for predicting user preferences for newly encountered contexts. Preference mediation was first studied with respect to adaptation to non-definable computational factors [Berkovsky 2008, Anand 2010], i.e. how to use ratings acquired by one recommender system in another. Berkovsky et al. [Berkovsky 2008] suggested fairly simple mediation methods such as computing similarity in ways that reflect correlations between contexts, while Anand and Bharadwaj [Anand 2010] proposed to modify the similarity measure with a genetic algorithm. The initial and target contexts in these works did not differ significantly, however, as both were databases of movie ratings. Blanco-Fernandez et al. [Blanco-Fernandez 2010], on the other hand, studied mediation in the case of significantly different contexts, a TV recommender and a tourist guide. In their system the fairly generic user interests were inferred from TV viewing histories and then tourist attractions belonging to the same categories were recommended, e.g. diving was recommended for users watching TV programmes about water sports. Thus this approach relied on the assumption that generic user interests are context-independent, which is not always the case. One more mediation method was suggested by Baltrunas et al. [Baltrunas 2012], the shifting of a preference vector learned from data on users who had already ranked items in both the initial and target contexts. This model was tested for tourist guides for a fairly large variety of context types, such as weather, company, budget, travel goal, etc., and the results showed that some context types influence the ratings of all the categories of attractions whereas the influence of some other context types depends on a certain category. In our UI test case [Publications S5 and IV] we aimed at adaptation to both similar and different contexts with no assumptions regarding the context similarity or context-independence of user interests. Shifting the preference vector is compared with modifications of the similarity measure and other approaches in Publications S5 and IV.

One common way of providing recommendations for groups is to combine the individual profiles of group members in various ways [Jameson 2007, Masthoff 2006, Yu 2006, Senot 2010, Salamo 2012]. This has many drawbacks, however: it relies on domain knowledge, it ignores the fact that a group is more than the sum of its members [Jameson 2007], and it fails in the case of groups of people with significantly different personal preferences [Yu 2006, Salamo 2012]. Accordingly, adaptation to groups can achieve better by pre-defining certain group types, for example by treating the degree of heterogeneity between group members as a context and selecting a reasoning method based on this degree. Sotelo et al. [Sotelo 2009] employed two degrees only: homogeneous and heterogeneous groups. Recommendations for the former were provided by combining individual preferences and those for the latter by CF. Shin and Woo [Shin 2009] employed three degrees and three strategies: 1) automatic item selection, 2) item recommendation, and 3) item category recommendation with explanations, helping groups to make their own decisions.

The modelling of relations between group members has also been proposed. Chen et al. [Chen 2008] suggested learning (with a genetic algorithm) the degrees of influence of group members on the group ratings, using data collected during group sessions and sessions of the subgroups. This approach requires fairly long-term use of the system in order to collect ratings from the subgroups, and due to the difficulty of acquiring such data, Chen et al. tested the proposed method on simulated data only. Thyagaraju and Kulkarni [Thyagaraju 2011] suggested avoiding data collection by employing designer-defined rules for estimating the social dominance of each family member based on his/her age, role (father, mother, etc.) and income, using this to predict preferences for each group member in the social context, and then combining these predicted preferences for use in CF-based predictions for groups, thereby differing from the common approach of using the preferences of group members in a “being alone” situation. Cultural diversity and individual differences make it difficult to define rules that are suitable for all families, however.

Adaptation to heterogeneous groups by means of negotiations was proposed in [Garcia 2014]: first User Agents of all group members make proposals according to personal preferences, then a Negotiator Agent combines the proposals based on pre-defined reasoning logic and sends the result to the User Agents that may accept it or make another proposal. The process continues until either agreement is reached or no new proposal can be made. Choices of the User Agents depend not only on individual user preferences, but also on their degrees of cooperativeness (i.e., whether the users mainly want to satisfy own tastes or ease an agreement) and concession tactics (i.e., how many preferences of other group members, differing from own preferences, the users accept at each iteration of negotiations). How the users would specify these parameters and whether they would accept the system suggestions, remains to be studied because the proposed system was not tested with real user groups.

Instead of modelling relations between group members explicitly, we have proposed learning the preferences of individuals and groups from observations of their choices [Publications S3, II, IV].

2.4.3 Ensembles

A combination-based ensemble employing context-independent base classifiers was used in TV recommender system in [Holbling 2010]. Here each ensemble member was trained on a single programme attribute (such as “genre”) and the weights of their outputs were adapted according to user feedback. Comparison of such an ensemble with its alternative for training a single model with respect to several attributes demonstrated that the ensemble adapted to individual tastes more accurately. In our work different classifiers were trained on different programme attributes, too, but their outputs were combined by voting or by means of fixed weights and then the best combination scheme was selected, because this approach is more robust to noise in the implicit feedback [Publication II].

Combination-based ensembles have also been employed for dealing with concept drift (namely, changes in users’ interests over time), but in these cases the base classifiers have frequently been re-trained on data bins of different lengths [Apeh 2013, Christou 2012]. Apeh and Gabrys [Apeh 2013], in their work on modelling buying behaviour, did not adapt a method to combine member outputs (they employed voting), whereas Christou et al. [Christou 2012], in their TV programme recommender system for individuals, employed two decision-making strategies on top of the base classifiers: at the beginning of each time bin the member outputs were combined by voting and at the end decisions were made by the most accurate classifier for the data in that time window.

Selection-based ensembles may also select a base classifier depending on data availability: for example, tourism recommender systems may provide recommendations for new users based on their demographic data and switch to collaborative filtering and/ or content-based recommendations when longer interaction histories of this user are obtained [Borras 2014]. Selection-based ensembles may also employ conventional dynamic classifier selection approach: to choose the best member for each data sample based on sample features. Zliobaite et al. [Zliobaite 2012] employed this approach for modelling buying behaviour. The ensemble included two members, a “moving average” and a regression tree, trained on sales data and context data (holidays, season and weather). The best member was selected for each product based on its features.

Another selection-based ensemble was proposed for a CF-based recommender system [Adomavicius 2005] in which the ensemble members searched for similar users using general (context-independent) user profiles and profiles created using different options for generalising the context (e.g. ranks acquired on other weekdays were also used for predicting preferences in a “Tuesday” context). This ensemble achieved good results because the training datasets for some contexts were too small and the user preferences did not significantly depend on certain context types. Instead of employing different user profiles, Baltrunas and Ricci [Baltrunas 2014] modelled each item as a set of fictitious objects, each one being the same item, but in a different context. An item was split into a set of fictitious items only if splitting resulted in more homogeneous sets of ratings than that in the initial set. Predictions were then made by a standard collaborative filtering approach either for the initial or the fictitious item.

2.4.4 Context as a feature

A fairly common approach to context modelling in CF-based systems is to include contextual similarity in the distance measure. Ahn et al. [Ahn 2006] suggested weighting the distance between users’ ratings by similarities between the contexts in which these ratings were provided. These contexts were pre-defined: users’ needs (utilitarian vs. hedonic), day of week and time of day. Pre-defined contexts were also used in a distance measure for dealing with changes in users’ interests over time, reflecting the order of the consumption of items and differences between consumption times [Rafeh 2012].

In other reasoning methods context cues can serve as latent variables or additional inputs. Thus pre-defined purchase goals served as a latent factor in Bayesian networks [Palmisano 2008], and pre-defined contexts (time, place, etc.) served as inputs to SVM models in a restaurant recommender system [Oku 2006]. In a TV recommender system for individuals, learning from their interaction histories [Da Silva 2012], time, user location and device contexts served as inputs to CBR (Case-Based Reasoning), MLP (Multi-Layer Perceptron), Naïve Bayes and decision tree methods. In our TV recommender system for families, both the time and the identities of the family members served as inputs to CBR and SVM [Publications S3 and II].

2.5 Adaptation in the affective computing domain

2.5.1 Research topics and open issues

The main open issue in affect recognition is adaptation to social context. This is an important problem because social rules and personal differences greatly influence the expression of emotions [Calvo 2010, Douglas-Cowie 2007, Zeng 2009, Vinciarelli 2012]. A person who is upset may scream and grimace in one context, for example, whereas in another he/she may remain silent as a matter of etiquette and his/her emotions can be recognised only from facial expressions [Wagner 2011]. Schuller et al. [Schuller 2010] performed a cross-corpus evaluation of an audio-based emotion classifier on six databases collected in different countries and containing spontaneous, induced or acted emotions. They compared the capabilities of the system for recognising emotional categories (e.g. joy, anger, etc.) and for distinguishing between positive vs. negative arousal and valence, and concluded that “performance is decreased dramatically when operating cross-corpora-wise”, mainly due to differences between ways of acting or spontaneously displaying emotions in different contexts.

Another important research topic is adaptation to historical factors (mainly previous emotional states). This is an important problem because emotions may last for a long time, e.g. an excited person may not calm down instantly. Adaptation to environmental and computational factors is not the main focus of research in this domain because databases are usually collected in controlled conditions. Task-related factors mainly influence affect recognition systems indirectly, via the choice of emotions to recognise and their representation (categorical vs. dimensional). Finding highlights in multimedia content, for example, may require recognising basic Ekmanian emotional states [Joho 2011], whereas e-learning applications are mainly concerned with the recognition of non-Ekmanian emotions such as the leaner's interest or boredom [Forbes-Riley 2004, Schuller 2010]. The choice of a categorical vs. dimensional model does not significantly depend on the goal of the application. Thus Schuller et al. [Schuller 2010], for example, distinguished between several levels of learner's interest/disinterest, whereas Forbes-Riley and Litman [Forbes-Riley 2004] employed a categorical model of learner's emotions.

The choice of a categorical vs. a dimensional model depends mostly on the affordable labelling effort. In view of the difficulty in understanding human behaviour, emotional data are usually labelled explicitly, and thus the effort required is among the main reasons why context-dependent emotion recognition is still only an emerging research topic [Zeng 2009, Vinciarelli 2012]. As the representation of emotions in dimensional models is not intuitive, annotators may need special training to do it [Zeng 2009]. Thus discrete intensity levels are fairly frequently assigned to emotional data instead of using a continuous labelling space [Gunes 2010, Zeng 2009]. In our affect recognition test case [Publication III] we focused on reducing the need for labelled data and effort involved in data labelling and hence employed discrete levels of emotional intensity.

2.5.2 Model selection

Course-grained adaptation can be performed by pre-defining typical situations and mapping them onto context-specific reasoning methods. The works [Tawari 2010 and Kotti 2012], for example, employed separate models for males and females, each trained on target context data. The SVM classifier used in [Tawari 2010] was trained on a generic feature set, whereas in [Kotti 2012] feature selection was also gender-dependent. Meanwhile, Nguyen et al. [Nguyen 2012] studied the recognition of head nods as social signals in dialogues and observed that nodding in response to someone differs from the head movements of a talking person. Accordingly, separate SVM models were trained for “user is talking” and “user is silent” contexts.

Finer adaptation to indefinable social situations was achieved by Caridakis et al. [Caridakis 2008] by means of model-level knowledge transfer, in which audio-visual records of subjects communicating with four artificial computer characters were used to create a new model for each temporal segment in which the user had a steady emotional state. Thus these models were both context and user-specific. Adaptation of fully supervised neural networks was performed by means of a gradient descent-based search for increments in network parameters that minimise the weighted sum of errors in the old and new data. This procedure was fairly lightweight because only small perturbations in the parameters were allowed and the activation function of the network neurons was linearized with a first-order Taylor series expansion. The contexts did not differ significantly in this work, however, as all the data were recorded in the same lab and the test subjects were not communicating with real humans but with computer characters, which do not fall under exactly same social rules. Hence the emotional expressions of the test subjects were not very intense [SAL]. Our work, which was aimed at adaptation to notably different contexts, employed a differential evolution algorithm for transferring the HMM models and permitted fairly significant parameter changes [Publication III].

2.5.3 Ensembles

Ensembles have been employed in this domain for dealing with the problem of missing inputs due to personal differences in expressing emotions or to algorithm failures [Wagner 2011]. The ensembles took the form of a classifier cascade, evaluating all the base classifiers first on training data and then appointing the most accurate classifier for each class as a “specialist” for its class, the next most accurate as the “second best specialist”, etc. The classes were then ordered from the worst classified (most difficult) class to the best one. At the fusion stage a sample was first sent to the specialist for the most difficult class. If this specialist classified the sample, the process stopped (this approach prevented the assigning of too many samples to the dominant class). Otherwise the sample was sent to next specialist, and so on until it was classified. If the “specialist” for the corresponding class required a missing modality, the sample was sent to the “second best specialist”, etc. The resulting ensemble was compared with several other fusion methods (Weighted Sum, Product, Weighted Voting etc.), dealing with missing data by recalculating the weights to sum up to one. All strategies used in the experiments achieved similar accuracies.

Ensembles have been employed also for adaptation to personal differences in expressing pain. Chen et al. [Chen 2013] suggested to train (with AdaBoost algorithm) a separate pain recognition ensemble for each person in training dataset. All members of all resulting ensembles then constitute a pool, and adaptation to a new person is performed by optimising weights of classifiers in this pool by AdaBoost algorithm. Optimisation aims at minimising error rate for the target person and is very quick because new base classifiers are not trained in this stage: AdaBoost algorithm only selects most appropriate ensemble members from existing ones. This lightweight adaptation was compared with two other approaches to building context-specific models:

- training of a person-specific model from scratch using data for the target person only;
- training of a person-specific model from scratch using mixed data for the target and all other persons.

In the tests the ensemble adaptation took 0.16 minutes per subject on average, while training of a person-specific model from scratch using data for the target person only required 2.6 minutes. The person-specific models were also notably less accurate then the person-specific ensembles when number of labelled training samples per target person ranged from 10 to 50. Accuracy of a person-specific model, trained using mixed data, was similar to that of the ensemble when amount of labelled training data per target person ranged from 10 to 25 samples, but considerably lower when training dataset per target person included 50 and more samples. Training of a person-specific model from scratch using mixed data required notably longer time, however: 14.3 minutes per subject on average. Unfortunately it is not explained in the paper whether ways to express pain differed significantly between the subjects and if so, whether the proposed method correctly recognised pain of the subjects, most different from the others.

2.5.4 Context as a feature

A past emotional state will fairly often serve as a feature. Schuller et al. [Schuller 2009] and Forbes-Riley and Litman [Forbes-Riley 2004] used past emotional data as additional input features in SVM and AdaBoost algorithms, respectively, and such data have also been used in Long Short-Term Memory Recurrent Neural Networks (LSTM) [Wollmer 2010, Metallinou 2012] and in HMM [Metallinou 2012]. In our work past emotional state served as a feature in HMM and SVM [Publication III].

Past events can serve as features, too. López-Cózar et al. [Lopez-Cozar 2011] developed an emotion recognition method for spoken dialogue systems in which two types of pre-defined past events were recognised: dialogue acts such as repetition and rephrasing (needed when a system cannot understand a user immediately) and lexical expressions (for example, when a user says “no, I said…”, one may deduce that the user is correcting a system mistake and it does not make him/her any happier). The outputs of these classifiers, in the form of probabilities of emotional categories, were combined with probabilities estimated using acoustic and prosodic features. This was done in two stages, each employing a non-trainable fusion method, e.g., voting or taking the average or product of the probabilities. Similarly, dialog acts served as context in two-stage classifier in [Griol 2014]. At the first stage Multi-Layer Perceptron employed acoustic features to classify user state into two categories: 1) “angry” and 2) “doubtful or bored”, and at the second stage dialog acts were employed for rule-based classification of the latter into “doubtful” and “bored” states.

Environmental parameters can serve as nodes in graphical models. A dynamic Bayesian network in a system for emotion recognition in drivers [Li 2005] included nodes taking pre-defined discrete values and representing context (complex vs. simple road situation) and user characteristics (skills, physical and mental condition). This approach allowed the interpretation of video cues (e.g. high vs. low gaze fixation) and audio cues (answers to questions) in a context and user-dependent manner. Data collection was avoided by specifying the network parameters by hand.

2.6 Adaptation in the user interaction domain

2.6.1 Research topics and open issues

The main open issue in this domain is adaptation to context-dependent personal preferences. It is still rare to adapt interfaces to both personal preferences and contexts, although it was shown that users appreciate such a possibility at least with respect to menu customisation [Bohmer 2010] and virtual environments [Octavia 2011]. Context-independent adaptation to personal preferences is similarly uncommon, despite the fact that when users are allowed to customise interaction they appreciate opportunity to customise and choose not only most practical options for them, but also the most interesting ones [Reis 2008]. As application designers are not likely to predict which options may be most interesting for users in different contexts, interaction personalisation is considered an important future direction of research [Dumas 2009, Turk 2014]. User preferences can be acquired implicitly via tracking customisation choices, or explicitly by asking users to rank options or perform certain tasks and evaluate their speed.

The majority of research in this area has focused on adaptation to computational factors (such as screen size and capability of devices to provide information via a certain modality, e.g. audio) and adaptation of mobile devices to task factors (such as finding a nearest restaurant or answering a call vs. answering a text message) and the environment (e.g. light and weather, as wearing gloves or a hood may hinder the use of certain modalities), while adaptation to social factors (e.g. the fact that speech interaction is not common in public) was only suggested as an important direction of research [Ronkainen 2010]. Recent works studied also adaptation of the levels of obtrusiveness of interaction to user preferences [Gil 2012] and adaptation of the levels of system autonomy to quality of contextual data, for example, acting on behalf of a user if the system is certain about his/her context vs. displaying suggestions instead of acting if the system is less certain [Hossain 2013]. In the majority of cases, however, adaptation follows either user-provided or designer-generated adaptation rules. Hence a recent review [Turk 2014] suggested that interaction adaptation requires fundamental improvements, based on machine learning techniques.

2.6.2 Model selection

Gajos et al. [Gajos 2010] employed constraint-based optimisation for adapting GUI to different screen sizes, taking into account user preferences and abilities, e.g. the special needs of motor-impaired users. For example, when a user clicks on interface elements of various sizes the system can trace the user's speed and generate size constraints. Elicitation of these constraints nevertheless requires fairly long and complicated interaction histories, so that able-bodied participants had to perform tasks for at least 25 minutes in the tests and motor-impaired participants for 30-90 minutes [Gajos 2008]. The acquired preferences were then assumed to be valid for different screen sizes, but this assumption was tested only for individual usage and for fairly similar tasks (controlling light intensity, ventilation and audio-visual equipment in a classroom).

Due to the difficulty to obtain user preferences implicitly, in [Macik 2014] utilities of different GUI options (e.g., different font sizes) were partially obtained from users and partially specified by system designers. These data were used for cost-based interface adaptation to different platforms.

Kong et al. [Kong 2011] optimised interfaces by maximising the sum of the scores for their elements. Here a greater variety of contexts and interaction modalities were considered, as the contexts included weather, light, noise, motion, screen size, keyboard type, etc., and the modalities included eye tracking, gestures, audio, video, vibration, etc. User preferences for interacting with a social networking application were acquired by asking the users to assign numerical preference scores manually for the various interface elements in different contexts. This is a more tiring and error-prone approach than selecting the most appropriate elements from available options, as was done in our solution [Publication IV]. Nevertheless, Kong et al. [Kong 2011] did not attempt to predict preferences in new contexts.

2.6.3 Context as a feature

Certain user characteristics can be included in input feature vector instead of user identities. Capuano et al. [Capuano 2015] suggested to use "big five" personality traits of individuals as input features to a neural network, trained to adapt interaction style (e.g., dialog-based, browsing etc.). This approach reduces the need in individual interaction histories, but requires additional data to obtain personality traits. In [Capuano 2015] these traits were obtained via analysis of posts, written by the test subjects in social networks.

3. Lightweight adaptation studies

The present work focuses on building situation-adaptive multimodal fusion models and on overall system design. The analysis of individual modalities lies beyond the scope of this work. As runtime feature selection is computationally fairly expensive, it is assumed below that the types of data and context cues are defined at the design stage and exact sets of cues can either be defined at the design stage or read from the data during the runtime.

3.1 Application factors and adaptation choices

The adaptation experiences obtained in the four test cases described below are gathered together and integrated with information from the related literature in Publication VI. The paper first summarises the approaches proposed for adaptation to newly emerging situations in the different domains and discusses their applicability to various tasks. This analysis enables descriptions to be given of the major design choices, so that the characteristics of the applications that influence the adaptation design most significantly can be identified. The paper then recommends how those characteristics can be taken into account in adaptation design.

Contextual factors, especially high-level ones, are usually non-discriminative with respect to the classification task in hand, i.e. they do not directly help in classifying data. Instead, situational changes cause certain changes in the primary data, e.g. in that humans usually express feelings more freely in conversations with persons close to them than with officials. Publication VI suggests categorising changes in primary data cues as follows:

- *Meaning* changes: the same input cues need to be interpreted differently in different contexts.
- *Influence* changes: the same input cues may differ in importance depending on the context.
- *Accuracy* changes: the same input cues may be recognised more or less reliably in different contexts.
- *Availability* changes: the same input cues may be abundant in some situations and missing in others.

Based on the literature review and the experiences gained from the four test cases, Publication VI suggests that the following application features should be considered important:

- *Adaptation time*: almost instant, a few minutes, lifelong learning;
- *Permitted degree of user control vs. application control*: whether users are allowed to re-train the system or not;
- *Expectations regarding changes in input cues* due to situational changes: meaning, availability, influence, accuracy;

- *Costs of data acquisition vs. data quality*: whether users and classifiers can significantly benefit from explicit human effort or not, and whether implicit interaction can be reliably interpreted or not;
- *Variability of situations*: whether an application is likely to encounter a few *a priori* unknown but fairly stable situations or many diverse situations, and whether these situations can be defined or not (the latter will be referred to below as "indefinable situations").

Publication VI also states that the most important design choices to make are the following:

- *Adaptation type*: 1) model selection, 2) ensemble, 3) context as a feature;
- *Interaction type*: 1) implicit, 2) explicit, 3) a combination of both;
- *Data usage type* (transfer of knowledge acquired in other situations): 1) none, 2) using a dataset acquired in other situations, 3) using a model trained on a dataset acquired in other situations;
- *Runtime training type*: 1) none, 2) a standard algorithm, 3) customised modification of parameters;
- *Training supervision*: 1) none, 2) partial, 3) full; and
- *Reasoning methods*: 1) lazy, 2) fixed (e.g. majority voting), 3) trained (graph-based or other).

These design choices, which depend on application specifics and on each other, determine how models for new situations can be learned at the runtime stage, so that if the adaptation type is the use of context as a feature, for example, then full re-training of the model will be required each time a new situation is encountered, whereas if the adaptation type is model selection and knowledge obtained from other situations is used in a form of models for these situations, then model-level knowledge transfer (i.e. modification of the parameters of existing models) can support learning as many new situations as desirable. The major effects of contextual factors on adaptation design are illustrated in Figure 4.

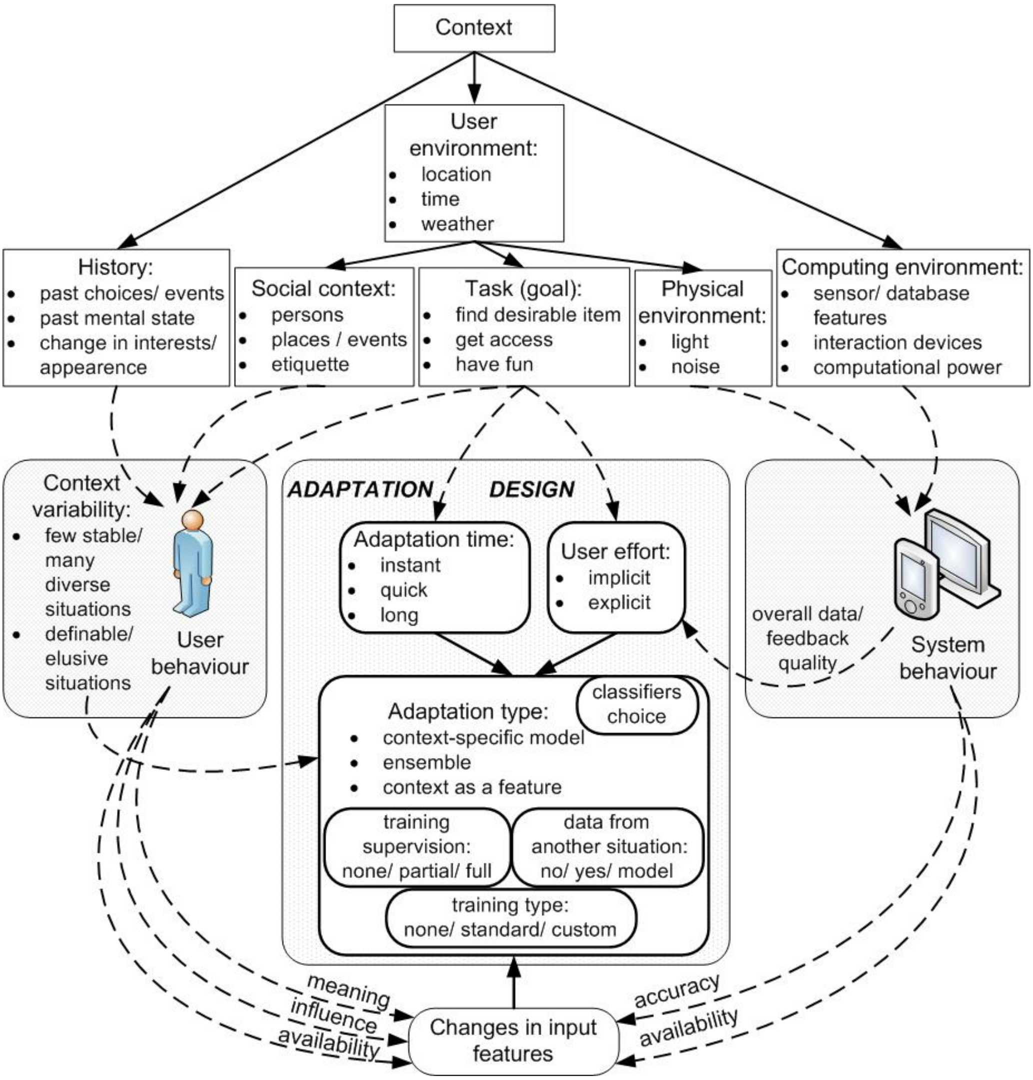


Figure 4: Influence of context on design choices (common effects).

Publication VI recommends that the interaction type should be chosen based on the expected costs of data acquisition and data quality. The implicit interaction should be employed, if possible, when quality of primary data cues is rather low. For example, if a certain cue is important for users, but not recognised by a system, accurate models cannot be built despite all the explicit annotation efforts; instead, the latter will only annoy the users. Explicit efforts are necessary if interpretation of implicit interaction data depends on context: e.g., if just the same user actions may denote a positive feedback in one context and neutrality in another context. Acquiring explicit interaction data can be suggested also for the changes in meanings of the input cues because several studies suggest that quick learning of new meanings may be difficult without explicit human supervision. Otherwise, for choosing between the implicit and explicit interaction we recommend to consider, first, to what extent both labelling time and a time interval when the labels are used by the system match the users' goals and, second, how the UI design and social rules affect quality of implicit data: for example, the implicit feedback may be useless if the users click on nearly every link because of insufficient link data and if the users adjust their choices to be polite instead of following their own desires.

Then the adaptation type and data usage types should be decided on in the light of another two of the most influential application features, the variability in situations and expectations regarding changes in input cues brought about by situational changes. Training an own model for each context is one of the most adaptive approaches. It can be applied to a broader range of cases than those having been studied up to date and handle primary data cues, emerging at runtime, provided that their types and the use of different types inside the models can be predefined and the models can be trained at runtime. The classifier ensemble, where all members are trained on the target context data, is even more adaptive, but less lightweight than training an own model for each context. Accordingly, employing such ensembles to adapt to contexts, emerging at runtime, is feasible mainly when large datasets are acquired naturally in the course of using an application, e.g., when implicit interaction data can be collected. In other cases, either the classifiers learnable from very small data sets are to be included into the ensembles, or other ensemble types should be chosen. Combining outputs of the base classifiers can be recommended only when all these classifiers are sufficiently context-independent or trained on target context data. Combining outputs of the base classifiers, trained on the data of other contexts, may result in low accuracy if the majority of these classifiers do not suit well for the target context. In other cases the selection-based ensembles should be employed: they can handle both similar and dissimilar contexts if different members are optimised for different contexts, and they outperform individual classifiers when amount of training data is not abundant.

Models, using context descriptors as features, may be trained for each context or be included into ensembles. Using context as a feature in a single classifier, trained on the data of many contexts, is an approach suitable mainly in cases of changes in the input cues' influences.

Regarding the data usage types, using both labelled and unlabelled target context data could be beneficial when both could be obtained. Benefits of using unlabelled data of the non-target situations in the lightweight adaptation have not yet been demonstrated. Training the context-specific models on the merged data of different contexts may hinder the adaptation to notably different contexts. Therefore, the use of raw data of other contexts can be recommended only if the chosen knowledge transfer approach requires such data. Using trained models of the non-target contexts can be recommended for faster adaptation. To use the target context data only can be recommended when the training datasets are not too small (e.g., if implicit interaction data or unlabelled data are available) or the applications are supposed to be long-term user companions and thus need to gain the user's trust by avoiding data sharing and reasoning errors.

Publication VI suggests that adaptation to changes in meanings of input cues is best of all achieved via training a separate model for each context and employing either model-level or no knowledge transfer. Adaptation to changes in the cues' availability is best of all achieved via training a separate model for each context or by treating cue availability as a feature in reasoning methods that handle the missing cues without re-training, e.g., discrete HMMs, weighted sums, voting etc. In both cases Publication VI recommends to train models on the target context data only. Choice of an approach to adapt to changes in accuracy or influence of input cues strongly depends on expectations regarding variability of situations. If an application is likely to encounter large number of situations, fairly lightweight approaches should be chosen, such as selection-based ensembles or model selection. Otherwise more accurate approaches should be chosen, such as use of the target context data only.

Publication VI also describes how other matters such as the choice of runtime training type may depend on the above decisions. Model-level knowledge transfer and old knowledge preservation usually require custom algorithms, e.g., some additional constraints on the model parameters may be added. Custom adaptation, such as with evolutionary algorithms or various re-weighting schemes, is usual in selecting and/or combining the ensemble members, too. Standard training is more common in other cases due to its easiness for developers.

The four case studies will be described below, with notes on how the application features that were identified determined the adaptation design in each case.

3.2 Software framework

Details of the software framework developed for the various experiments can be found in Publication V. The runtime adaptation of class-level or decision-level fusion required implementation of the following functionalities:

- handling changes in the availability, accuracy, influence and meanings of the input cues;
- choosing between adaptation types (context as a feature, ensembles and model selection);
- learning new situations from scratch vs. modifying the parameters of existing models with the help of user feedback.

Reliance on user feedback in learning also implies a need to interpret implicit feedback when this is obtained. Furthermore, as it is not feasible to expect error-free feedback, the machine learning algorithms employed should be noise-robust. A block diagram of lightweight situational adaptation used in the test cases is presented in Figure 5.

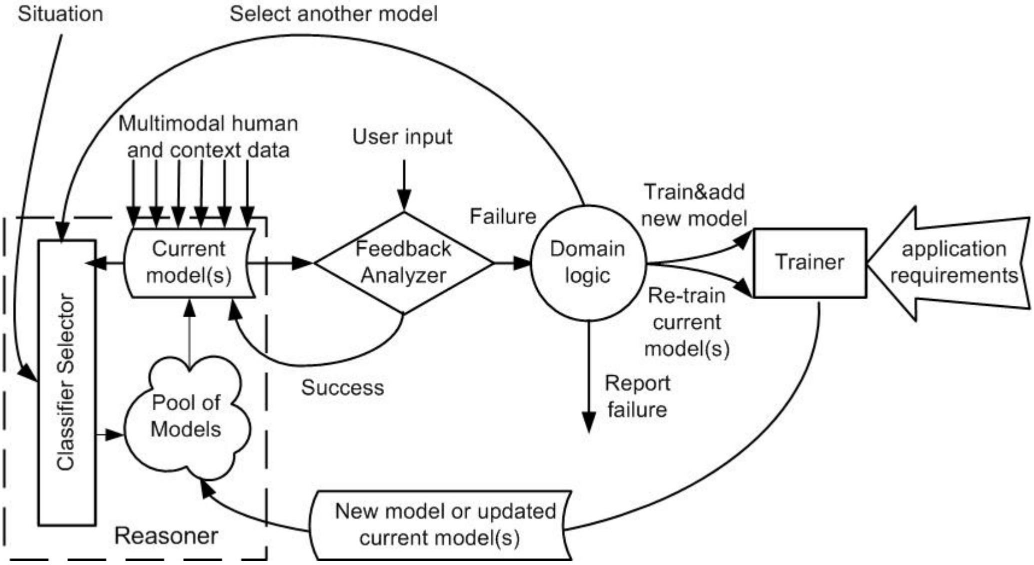


Figure 5: SW components of lightweight runtime adaptation.

The SW blocks in Figure 5 have the following responsibilities:

- *Pool of models*: includes all available models and their metadata.
- *Classifier selector*: retrieves current models, i.e. those most appropriate to the current situation. Models can be selected according to their metadata (e.g. by situation name), situational parameters (e.g. within a range of signal-to-noise ratios) or accuracy. We assumed that situation recognition would be performed outside the classifier selector, so that the SNR (signal-to-noise ratio) can be detected by the current audio analysis model, and a situation name can be provided by external context recognition components or entered manually by the user. Although user-specific models can be selected in the same way, our work does not regard model selection according to user ID or family ID as situational adaptation because our work is concerned with more intricate adap-

tation aspects. This component was implemented in a fairly generic way and employed in all the test cases.

- *Reasoner*: realises the data flow used for training and fusion, returns the "final" classification result and the degree of confidence in it. This component was implemented in a fairly generic way and was employed in all the test cases, but some parts were modified in task-specific ways.
- *Feedback Analyzer*: obtains user interaction data and user requests to select another model, to re-train existing models or to learn a new model. In the case of ensemble-based adaptation this component will also evaluate the accuracies of the ensemble members. This component was implemented in task-specific ways in all the test cases. In the TV recommender system it was fairly complex, as it propagated user choices for TV programmes to the lower-level models for programme genres, channels, etc., created new positive and negative training examples from current recommendations and user choices, evaluated the accuracies of all the models in the pool and triggered re-training when a sufficient number of new training examples had been obtained. In the biometric domain this component was very simple, as it only triggered the retrieval of another model when needed, as was also the case in the UI adaptation domain, where it collected interface settings for the different modalities and evaluated the accuracies of the ensemble members. In the emotion recognition domain it was somewhat more complex, collecting user-provided labels, assigning them to the input data within a short time window and triggering the training of a new model when requested by the user.
- *Domain Logic*: the component which knows whether runtime training is allowed and if so, whether a new model should be added or existing models should be re-trained. This logic was implemented in task-specific way for each test case.
- *Trainer*: the component responsible for training new models and updating existing models according to the application requirements (e.g., maximum allowed false acceptance ratio of user verification). This component was implemented in the test cases for the chosen classifiers. Due to the requirement for handling noisy labelled data, our work employed SVM and HMM as trained classifiers and neighbourhood-based reasoning (kNN and CBR), voting and other heuristic methods as lazy classifiers. In some test cases MLP (Multi-Layer Perceptron) and Weighted Sum were also employed as trained classifiers, while in others a differential evolution method was employed for adapting weights in kNN and for modifying HMM parameters in customised ways. Standard training procedures were always employed for SVM and MLP.

3.3 Case 1: biometrics

As the literature review shows, increasing user convenience is an important goal for next generation biometric systems, but explicit verification is still the most common solution. This means that access is granted for a long time after the only successful verification, which allows the authorized user to be replaced by an impostor, as it is often the case with mobile phones. We proposed in 2006 to increase the security of mobile devices by means of an unobtrusive form of user verification based on gait and voice data [Publication S1], since the problem of user convenience had been largely ignored in research into biometric systems. As the use of unobtrusive modalities does not allow the achievement of low false reject and false accept rates simultaneously, we then proposed in 2007 that mobile devices should be protected using a cascaded system in which implicit verification is performed first and explicit interaction is required only if this implicit verification fails [Publication S2].

In Publication I we suggested and compared several ways of maintaining the desired security level while reducing the amount of explicit interaction at the inference stage.

3.3.1 Application scenario

The cascaded verification proposed in Publication I is illustrated in Figure 6. The verification suggested in Publication S2 employed a similar scheme, but with different modalities.

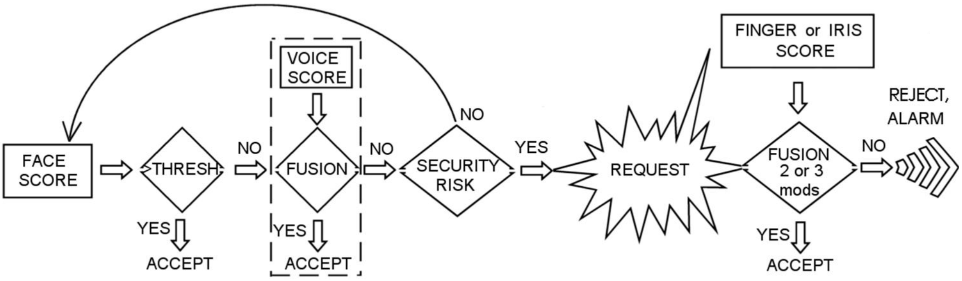


Figure 6: Verification scenario for two explicit and two unobtrusive biometric modalities [Publication I]. If the user is silent, fusion of the face and voice cues is skipped.

Adaptation to the following situational changes was studied in this test case: 1) changes in security requirements (e.g. paying a small parking fee vs. a significant money transfer), 2) changes in the availability of biometric modalities (e.g. a silent vs. talking user or a user preferring the iris modality over fingerprints), and 3) environmental changes (e.g. in background noise, such as low vs. high signal-to-noise ratio (SNR) for the voice recognition modality). The scores for the various modalities served as input cues to multimodal fusion module (a score is a degree of belief that the user is a genuine one).

This scenario has the following specifics:

- *Context factors to adapt to*: task factors (security level and data availability) and environmental factors (noise);
- *Adaptation time*: almost instant, as users usually want to access the applications very quickly;
- *Permitted degree of user control*: none, as allowing users to re-train the models may be insecure;
- *Expectations regarding changes in input cues*: context may affect the accuracy and availability of certain biometric modalities, and their influence on the recognition result may change;
- *Costs of data acquisition vs. data quality*: implicit data could be sufficient for low-security requirements but not for high-security ones; and
- *Variability of situations*: many distinct and definable situations exist, as it is sufficient to consider several security levels and several noise levels.

3.3.2 Classifier design

Context-independent components for user verification via face, voice, fingerprint, iris and accelerometer-based gait data were employed here. Every single modality component produced a score, and multimodal fusion was adapted to pre-defined situations: sets of available biometric modalities and several discrete security or SNR levels. The following design choices were made:

- *Interaction type*: implicit, plus explicit when implicit data do not suffice;
- *Adaptation type*: model selection due to significant differences between situations;
- *Data usage type*: raw target context data due to the availability of large datasets at the design stage;

- *Runtime training*: none, due to security requirements;
- *Offline training supervision*: full, due to data availability at the design stage; and
- *Reasoning methods*: cascade of SVM, MLP or Weighted Sum.

The system functionality is illustrated in Figure 7. The multimodal fusion models were selected on the ground of their metadata or context values. The system allowed the runtime addition of models for new situations, for which purpose the application designers would simply need to train a new model and provide its metadata (security level, range of SNR values and list of biometric modalities it can handle). Training would not affect any existing models, as the models for all contexts are trained independently of each other.

The training of cascaded systems capable of satisfying different security and data availability requirements is not yet a fully studied problem. In a parallel multimodal system (which uses all the biometric modalities at once) different ratios between the False Acceptance Rate (FAR) and False Rejection Rate (FRR) can be selected, since high-security applications require low FAR and have to accept higher FRR, while low-security applications require low FRR and have to accept higher FAR. In a cascaded system the applications entail a trade-off between FAR, FRR after the unobtrusive stage and FRR after the last stage. Thus several ways of creating and training fusion models for such a cascaded system were explored, including the suitability of different classifiers for high and low-security requirements. The following adaptation approaches (the classes of the approaches are described in more detail in Publication VI) were compared:

- *Context-specific models and data*: fusion of gait and voice scores by the Weighted Sum method, with weights calculated based on the accuracies of the modalities in the target context [Publication S1];
- *Context-specific classifiers*: cascaded system performing fusion of the implicit modalities first and requesting the explicit modalities only if the implicit verification fails. Each set of modalities has its own model trained for it, and each security level is trained separately from the other models [Publications I and S2].

Both approaches are considered lightweight, because they require little user effort at the inference stage and do not require runtime training. The design time training is not lightweight from the point of view of its computational resources, however.

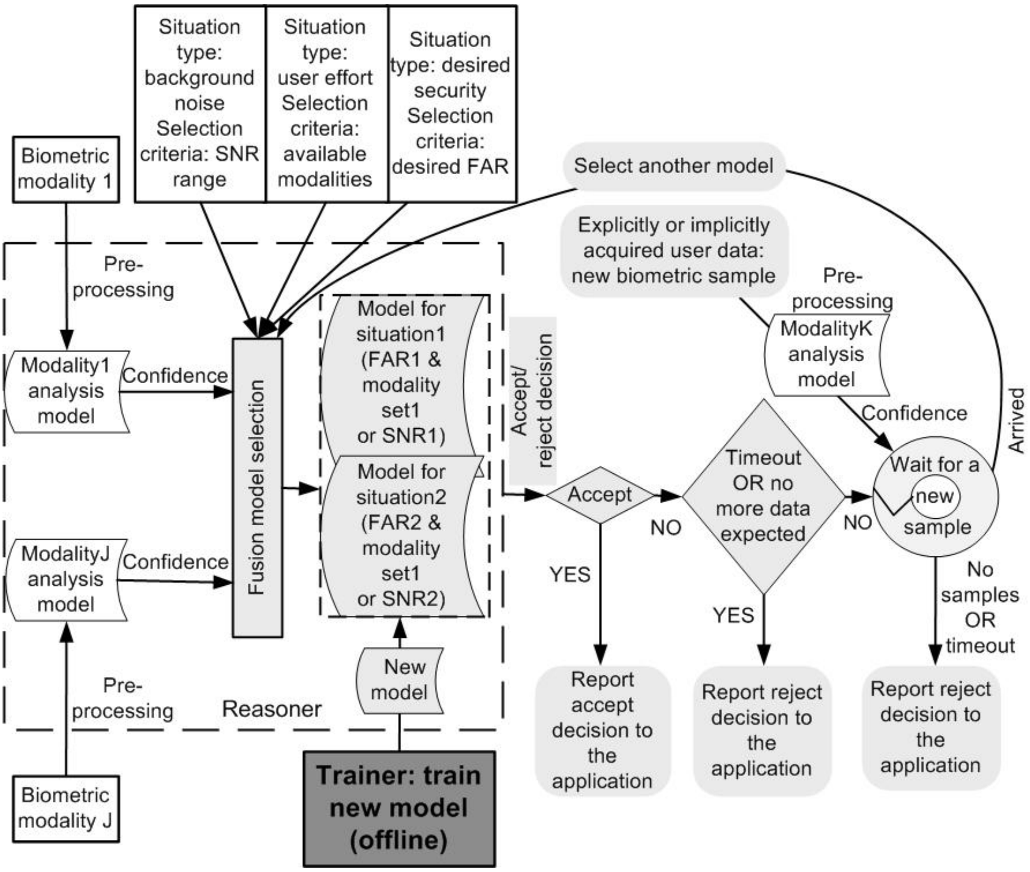

Situation type: background noise Selection criteria: SNR range
Situation type: user effort Selection criteria: available modalities
Situation type: desired security Selection criteria: desired FAR
Biometric modality 1
Select another model
Explicitly or implicitly acquired user data: new biometric sample
Pre-processing
Pre-processing
ModalityK analysis model
Modality1 analysis model
Confidence
Fusion model selection
Model for situation1 (FAR1 & modality set1 or SNR1)
Model for situation2 (FAR2 & modality set1 or SNR2)
Accept/ reject decision
Confidence
Arrived
Accept
NO
Timeout OR no more data expected
NO
Wait for a new sample
ModalityJ analysis model
Confidence
YES
YES
No samples OR timeout
New model
Pre-processing
Reasoner
Report accept decision to the application
Report reject decision to the application
Report reject decision to the application
Biometric modality J
Trainer: train new model (offline)

Figure 7: Lightweight adaptation for verification: grey colour indicates components developed for the multimodal fusion experiments.

3.3.3 Experimental results

Three datasets were used in the experiments: 1) gait and voice data for 31 subjects [Publication S1]; 2) gait, voice and fingerprint data for the same 31 subjects plus 19 more subjects [Publication S2]; and 3) voice, face, fingerprint and iris samples for 150 subjects [Publication I]. The experimental results confirmed the feasibility of the proposed lightweight adaptation at the fusion level, since despite the fairly poor performances (EER above 10%) and high overlap of errors in the unobtrusive modalities, the fusion models succeeded in adapting to the varying environmental conditions, the availability of biometric modalities and the different security levels. Voice-only user verification performed fairly badly under noisy conditions, for example, so that its EER exceeded 40%, but the noise level-adaptive multimodal system achieved an EER of 9–12% depending on where the mobile device was being carried [Publication S1]. The proposed cascaded fusion, where fingerprint samples were needed only when fusion of the gait and voice data failed, was able to satisfy security requirements of FAR ≤ 1% and to achieve an overall FRR of 3% or less, while requiring explicit effort in 10–60% of cases, depending on the noise level [Publication S2]. Thus the cascade allowed significantly higher accuracy to be achieved without any significant increase in explicit interaction efforts. In tests on the database of 150 users the proposed cascade satisfied a security requirement of FAR ≤ 0.1% when explicit interaction was required only in 35% of cases. These results cannot be compared with state-of-the-art reports because our system goals and databases were totally different.

3.3.4 Discussion

This research was aimed at instant adaptation to distinct situations that were pre-defined at the design stage, as runtime training was not permitted for security reasons. Thus training was performed at the design stage, when the obtaining of fairly large training datasets does not burden the users. Runtime adaptation was performed by selecting one of the existing models, although the system configuration easily allowed for the addition of models for new situations during the runtime. This approach is suitable for cases where the number of situations is limited, and it can also handle overlapping situations (e.g., by voting between equally suitable models), although we did not employ any overlapping situations in our work.

The selection of situations and domain specifics allowed us to explore two adaptation approaches: "*context-specific models and data*" vs. *"context-specific classifiers"*. The tests demonstrated that both approaches require data and computations, even though the former approach was employed in one of its simplest and most popular forms: a Weighted Sum with weights dependent on the accuracies of the modalities in the target context [Publication S1]. Nevertheless, it is necessary to iterate the training data for each situation for the same number of times as there are modalities. Thus such training requires time when the number of situations is large, especially for large datasets. When using *context-specific classifiers* the number of iterations of the training data was greater, but not dramatically so. Each stage in the cascade was trained independently of the others, because joint training was slower and the resulting accuracy was no higher [Publication I]. Thus each situation required training a separate model for each stage of each cascade configuration, but the overall number of configurations was reasonable and training of the stages employing just one modality was very easy.

The experimental results show that the more complicated approach of using *context-specific classifiers* is worth the effort, provided that the training time and data allow the investment of such an effort. In Publication I the accuracies of the proposed cascaded fusion were compared with those achieved with parallel fusion, i.e. in which all the available modalities are used simultaneously. Parallel fusion was also implemented in the form of *context-specific classifiers*, with training a model for each set of modalities. In tests with the dataset, containing scores of 150 users for four biometric modalities, the accuracies of parallel fusion were slightly higher than those of cascaded fusion, but the differences were statistically insignificant [Publication I]. Thus employing cascaded fusion instead of the conventional parallel fusion allowed a notable increase in user convenience without reducing the error rates. An attempt to reduce user effort by means of a parallel classifier would allow the number of pre-defined situations to be reduced and accordingly shorten the training time. However, this approach would require training a model for each security level that was capable of handling missing data, and unobtrusive verification would require dealing with one or two missing modalities out of three. As shown in the literature review state-of-the-art approaches to handling missing data within a single model do not cope very well with high percentages of missing modalities, and therefore such an approach would probably have resulted in lower accuracy than that achieved with context-specific cascaded fusion.

Regarding the comparison of reasoning methods, the test results on data for 150 users show that although SVM training takes longer, it satisfies security requirements more accurately than does the Weighted Sum, as the SVM models kept the FAR within the specified limits, whereas Weighted Sum fusion exceeded these thresholds. On the other hand, Weighted Sum fusion achieved a lower FRR than SVM. Thus SVM is better suited for cases with higher security requirements, whereas the Weighted Sum approach is better suited to low security requirements. Accordingly, adaptation to both types of security requirements would call for a classifier pool containing SVM models for some security levels and Weighted Sum models for the rest, which would mean employing *context-specific classifiers* to a great extent. Since MLP training took longer than SVM training and its results were less consistent, MLP is not recommended for this kind of adaptation even though its average accuracy did not differ significantly from that of SVM.

The results of the study performed in the biometric domain thus suggest that when it is necessary to adapt to a reasonable number of distinct, easy-to-define situations and runtime training is not allowed, *context-specific classifiers* provide more advantages than other adaptation approaches.

3.3.5 Summary

The author's work was aimed at simultaneously increasing the user-friendliness and security of biometric systems, while the majority of researchers have tended up to now to study one or other of these aspects. The main novelty of the solution, proposed by the author, lies in:

- employing a flexible cascaded architecture for inference, in which each stage can yield a final decision and any stage can be skipped if this increases user convenience without violating the application requirements;
- employing a model selection approach for adaptation to different types of situational changes simultaneously; and
- employing unobtrusive interaction at the first stages in the cascade.

The most interesting results achieved in this test case include first of all how to train such cascade. Cascaded fusion architectures, employing fairly inaccurate modalities at the first stage(s) and permitting to skip any stage have not so far been common, and thus methods for training them have not been properly examined. The experimental results suggest that it is feasible to train each stage separately. Secondly, this test case demonstrated that cascaded inference can achieve accuracies similar to that of parallel fusion (i.e. the simultaneous use of all modalities), but with notably less explicit effort. Thus cascaded inference, employing unobtrusive modalities first, seems to be an interesting future direction for research into biometric systems. Thirdly, a fairly popular and simple adaptation method for determining the context-specific weights of modalities was compared here with the more complex concept of trainable cascaded fusion, leading to the conclusion that the training efforts were comparable but the results obtained with cascaded fusion were notably better. Thus more complex adaptation design is evidently worth the effort in cases where pre-defined contexts can be employed.

Regarding the main drawbacks of conventional context adaptation approaches (reliance on domain knowledge and significant explicit interaction efforts), the former issue is more important for systems adapting to situations emerging during the runtime, while the latter issue is essential for all interactive systems. Consequently, the proposed biometric verification system did not address the former issue, but it did allow us to reduce the need for explicit interaction in three ways:

- by varying security levels depending on the current access requirements, so that not-so-accurate unobtrusive modalities will suffice for low security requirements;
- by requesting explicit efforts only when unobtrusive verification is insufficient for the current security level; and
- by allowing users to choose an explicit modality.

3.4 Case 2: a TV recommender

As the literature review shows, the adaptation of recommender systems to social contexts is largely an unsolved problem. Earlier works concerned with TV recommender systems were targeted mainly at individual users, while recent research into TV and movie recommender systems for groups has been targeted mainly at groups of friends or acquaintances rather than families. Our work aimed at developing recommender systems for families, taking account of the significant age differences and close emotional connections between group members.

It was proposed in Publication S3 that group behaviour could be modelled implicitly by observing, which choices group members make together and separately. As seen in the literature review, the most common approach to adaptation in the case of groups is to combine individual preferences. This approach has several drawbacks, however. First, combining preferences does not work well in heterogeneous groups. Families are often heterogeneous groups, with members who have their own ways of resolving conflicts, and it is not feasible to employ the same designer-provided logic for combining contradicting individual preferences in all families. Secondly, combining preferences does not provide for cases where group choices differ from the choices which group members would make alone, whereas sharing experiences is a fairly common activity in families, e.g. parents may watch programmes together with their children which they would not choose to watch at all without them, and vice versa. The proposed adaptation via learning from observations naturally adapts the recommendations to family practices, which differ greatly between families.

Based on the results in Publication S3, Publication II suggested that adaptation to family practices could be achieved more accurately by the runtime training of a classifier ensemble in which each base classifier is trained on data for the target context only.

3.4.1 Application scenario

As TV viewing is a leisure activity, users should not be obliged to provide feedback on programmes. Moreover, it is not so straightforward to evaluate user satisfaction in a multi-user environment, as group members may adjust their feedback to the opinions or feelings of others. Thus our TV programme recommender does not require explicit user feedback, but is able to use it if it is provided. As not all families would like to share their TV viewing data because of privacy concerns, our recommender system uses only data on a single household.

In this scenario a situational change occurs when a recommender system is launched in a new family. This scenario requires adaptation to a fairly indefinable course-grain situation (family habits), and certain lower-level context cues can be recognised automatically, e.g. time and social context (presence of family members near the TV). Programme metadata, time and social context served as input cues to the multimodal fusion module.

This scenario has the following specifics:

- *Context factors to adapt to*: social and task factors;
- *Adaptation time*: long, because the unobtrusive learning of habits of each family without sharing data between families cannot be done quickly;
- *Permitted degree of user control*: any;
- *Expectations regarding changes in input cues*: contextual factors may influence users' preferences differently in different situations (e.g. the day of the week may be more important for working people than for retired users; similarly, in some groups one person may dominate, whereas in others the interests of all members may have to be equally respected); feedback may carry slightly different meanings (e.g. if some users viewed a TV programme up to the end, it may mean that they were really interested in it, whereas for others watching a programme until the end may denote only slight interest);
- *Costs of data acquisition vs. data quality*: implicit feedback is available in large quantities and is fairly reliable, but the acquisition of explicit feedback is not feasible, because: 1) TV viewing is a leisure activity, and 2) the scarcity of metadata on TV programmes would not allow accurate enough distinctions to be made between programmes, and thus explicit feedback would not be fully utilised; and
- *Variability of situations*: one fairly stable, but indefinable situation.

3.4.2 Classifier design

The following single modality components were available: 1) application usage logger (a component retrieving a chosen TV channel and timestamps denoting the start and end of viewing); 2) metadata fetcher (a component retrieving TV programme details based on the channels chosen and the timestamps); and 3) context logger: a component retrieving data on the presence of family members in front of the TV and converting the timestamps into semantic times (day of week and time of day).

The context was recognised via dedicated sensors in finer detail here than in the affect recognition and UI adaptation domains, where only high-level labels for situations were obtained. Thus the use of context cues as features was feasible, moreover that they were discriminative features: fairly regular dependences of TV programme choices on context were observed in many families. In addition, an ensemble of diverse base classifiers was used for more accurate adaptation to indefinable social contexts such as family habits. The following design choices were made:

- *Interaction type*: implicit, as explicit feedback cannot be fully utilised.
- *Adaptation type*: selection-based ensemble with re-training of the base classifiers. Re-training was possible because of the availability of implicit feedback in large quantities and the opportunity to train base classifiers at night, when the users were sleeping and would not be bothered by the re-training.
- *Data usage type*: target context data only, due to privacy concerns and the availability of implicit feedback in large quantities;
- *Runtime training*: full training for SVM, as computational time was not an issue. CBR is a lazy reasoning method;
- *Runtime training supervision*: fully supervised, as implicit feedback can be acquired in fairly large quantities;
- *Reasoning methods*: Case-Based Reasoning (CBR) and Support Vector Machines (SVM) using context descriptors (IDs of present family members and semantic time) as features. These methods are able to deal with noisy data and do not need large training datasets, so they can start providing recommendations fairly soon.

The chosen adaptation approach belonged to the "*Context-specific ensemble"* class. The system was designed for learning each family model from scratch, because the privacy concerns of the test subjects made knowledge transfer undesirable. The appearance of a new family member would require complete re-training of the system. Nevertheless, the adaptation is fairly light-weight, because 1) runtime training does not require user efforts, 2) the need for computational resources is reduced by employing a small number of base classifiers, so that the ensemble includes only four members.

The ensemble was built in the manner presented in Figure 8. First level: context-independent components for deriving TV programme metadata, context information and positive/negative examples of users' choices from the timestamps. Next level: multimodal fusion models for various TV programme descriptors (name, genre and channel). At this level social and time context descriptors served as input features to CBR and SVM classifiers. Thus the exact sets of input features and models were family-dependent, but training and reasoning was performed in every family in exactly same (unified) way. The SVM models were re-trained during the runtime. Training took place at night, once a sufficient amount of new data had been collected, so that the training times would not annoy the end users. Models at the next level employed fixed combination rules for building lists of recommendations from outputs from all the name models, genre models and channel models.

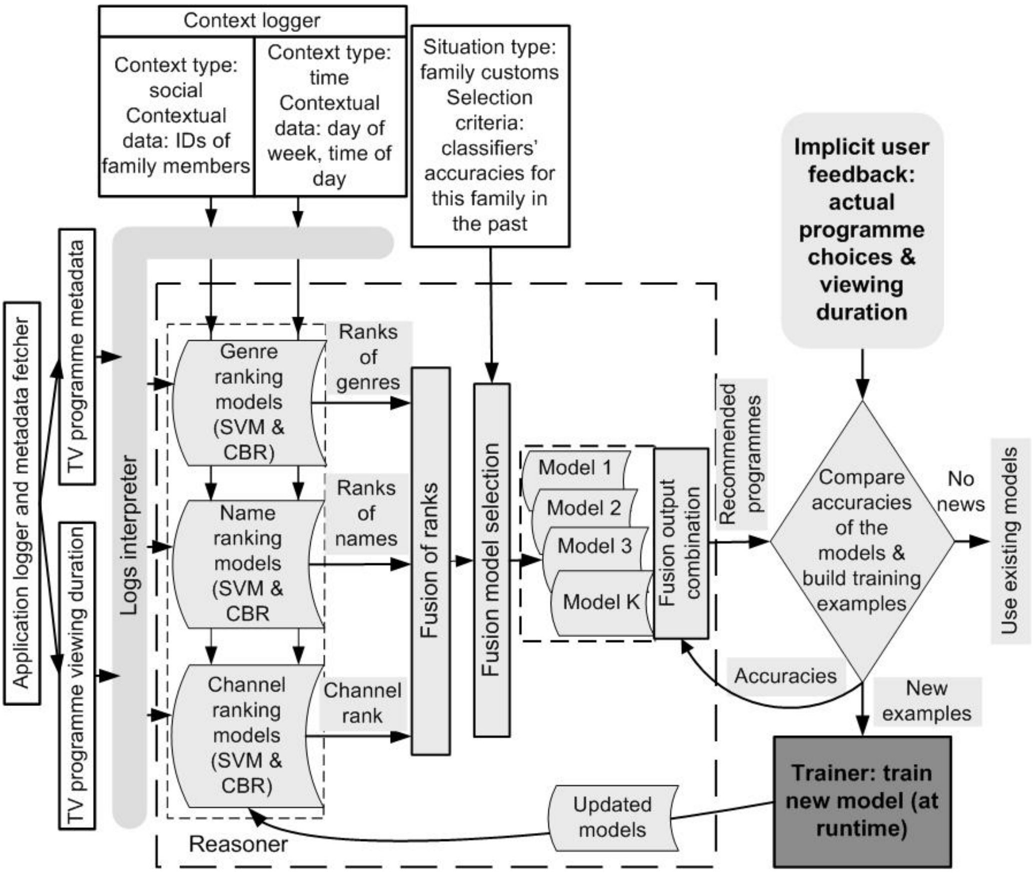

Context logger
Context type: social Contextual data: IDs of family members
Context type: time Contextual data: day of week, time of day
Situation type: family customs Selection criteria: classifiers' accuracies for this family in the past
Implicit user feedback: actual programme choices & viewing duration
Application logger and metadata fetcher
TV programme metadata
TV programme viewing duration
Logs interpreter
Genre ranking models (SVM & CBR)
Name ranking models (SVM & CBR)
Channel ranking models (SVM & CBR)
Ranks of genres
Ranks of names
Channel rank
Fusion of ranks
Fusion model selection
Model 1
Model 2
Model 3
Model K
Fusion output combination
Recommended programmes
Compare accuracies of the models & build training examples
No news
Use existing models
Accuracies
New examples
Trainer: train new model (at runtime)
Updated models
Reasoner

Figure 8: Adaptation of a TV recommender: the grey shading indicates components developed for the multimodal fusion experiments.

At the topmost level of the cascade a selection-based classifier ensemble was employed. This allowed more accurate adaptation to each family than would have been possible with multimodal SVM and CBR models only. Furthermore, this lightweight adaptation was performed each time recommendations were needed, whereas re-training of the ensemble members was performed only at night. Thus the ensemble allowed quicker adaptation to variations in daily viewing patterns.

3.4.3 Experimental results

The proposed unobtrusive modelling approach was tested offline on real life TV viewing data for 20 families collected over a period of five months. It achieved an average recall of 57% and an average precision of 30%. This recommendation accuracy is comparable with the accuracies of recommender systems for individuals that require explicit user feedback, despite the fact that group modelling is a more challenging task. Reasonable prediction accuracies for two-thirds of the families were already achieved after one month of incremental learning [Publication S3]. There were three families for whom the method did not exceed 50% recall among the top five recommendations even in the long run, whereas for three others the recall exceeded 70%, which is a good result for unobtrusive learning, i.e., learning by observations. The ensemble achieved a higher average accuracy than any of its members alone.

Unfortunately it is difficult to compare these results with other works because none of them used same data. Our results can be compared only with results of systems requiring explicit user effort. First, the CF-based system for individuals presented in [Adomavicius 2007] achieved 65–75% precision in the top five recommendations, while the precision of random recommendations was 35.6%. Our system achieved an average accuracy of 57%, but outperformed the random recommendations by factor of 3.3 on average (which means that our dataset was more challenging). Secondly, an algorithm for merging explicit user profiles, presented in [Yu 2006], achieved 75% recall at 45% precision, but members of the heterogeneous groups were not satisfied with the recommendations. The average precision of our system was about 30%, but it worked well for heterogeneous groups, in that 78% recall was achieved in a family whose members had quite different genre preferences. Thirdly, the TV programme recommender system for families presented in [Thyagaraju 2011] achieved fairly high precision in tests with three families, exceeding 60% in two, and ranging from 20 to 43% in the third. The reported numbers are not fully explained, however, e.g. the average number of programmes recommended is not quoted, so that it is unclear whether precision among the top two or top ten recommendations is being reported. Accuracy of random recommendations is not reported either, so that it is not possible to assess the difficulty of the dataset.

3.4.4 Discussion

Our main goal here was adaptation to family habits, which are indefinable situations. Situational changes caused changes in the influence of input cues, but no more dramatic changes. Adaptation was performed via runtime training on target context data that had been acquired implicitly. This approach is suitable for cases where the adaptation time is less important than the users' trust, and when the obtaining of a significant amount of training data does not require user efforts.

The adaptation type chosen here was a selection-based ensemble in which each member used context cues as features. Due to the need to reduce the training time, the ensemble included just four members, two of which used exactly the same base classifiers, just in different combinations, and the other two were lazy methods, so that re-training of the base classifiers was minimised. Nevertheless the ensemble adapted to fairly different families more accurately than any of its members, due to its greater robustness to noise (implicit feedback is fairly noisy) and its ability to benefit from a diversity of base classifiers. SVM is less sensitive to the use of non-discriminative features than CBR, and in this domain it is impossible to predict whether certain contextual cues will be discriminative or non-discriminative features, since in some families all contextual cues influenced programme choices, whereas in others the presence of one particular adult played a major role. On the other hand, CBR builds a decision boundary for each set of input features separately, whereas SVM builds one decision boundary for all sets. Thus it is easier for CBR to learn complex functions for representing inconsistent forms of group behaviour. One further difference is that CBR learns from positive examples only, whereas SVM is sensitive to the choice of negative examples. Consequently, CBR-based reasoning reflected the attitude of users who decided to spend time in front of the TV and tried to find the best programme from all those available, whereas due to the employed way of selecting negative examples, the SVM models for genres and channels reflected a goal-driven way of choosing activities, e.g., "now I want to see something funny and I am not in the proper mood to watch a drama, not even a famous one" [Publication S3].

Experiences from this study allowed us to suggest new diversity measures for context-adaptive ensembles. Most popular diversity measures for conventional ensembles are based on comparing the outputs of ensemble members on training data [Kuncheva 2004, Britto 2014], but they are not very suitable for context adaptation purposes because of possible significant differences between the training and test data. Our work suggests that diversity measures for context-adaptive ensembles should include 1) different sensitivity to non-discriminative inputs, 2) local vs. global learning of decision boundaries, and 3) modelling of different human decision-making strategies.

Regarding the use of context descriptors as features, the results show that this approach is feasible in the TV recommender domain because context descriptors are discriminative features: for example, in families with fairly stable viewing habits a fairly straightforward dependence of programme choice on time can be observed, while in families with members having notably different preferences a fairly straightforward dependence of programme choice on social context can be observed. In this respect recommender systems differ from affect recognition systems, for example, as context influences classifier output (emotion type) in more subtle ways in the latter. The tests also demonstrated that using context descriptors as features did not notably hinder the learning of context-independent preferences. Regarding the use of context descriptors as features in SVM and CBR, SVM was slightly more successful in the tests on data covering five months [Publication II], while CBR was slightly more successful in the tests on data covering two months [Publication S3]. The success of CBR and SVM was highly dependent on the family in question, however.

One important lesson learned in this domain is that the feasibility of using explicit feedback depends on the quality of the primary data. In our case no detailed metadata on TV programmes were available, and even the best-quality feedback could not help us to understand why certain programmes were preferred over others representing the same genre. Another lesson concerns use of implicit feedback. The maintaining of a pool of context-specific feedback interpretation models may allow more accurate adaptation, e.g. in that viewing a programme to the very end may denote a serious interest in it in one family, while in another the same behaviour may denote a moderate interest. Such a comparison of feedback models would significantly increase the amount of computation, however, since feedback models determine the selection and weighting of training examples, and thus each one would require full re-training of the classification models. Thus we have not studied this issue yet, nor are we aware that anyone else has studied it. This observation once again confirms the difficulty of adapting lower-level models, in that we did not test the computationally expensive training of interactive applications even though this is evidently feasible if performed during the applications' idle time.

3.4.5 Summary

The author studied here the specifics of recommender systems for families, taking account of the significant age differences and close emotional connections between family members, in contrast to the majority of researchers, who have studied adaptation to the individuals or groups of friends. As watching TV at home is a leisure activity, we decided that explicit interaction should not be required. The main novelty of the solution, proposed by the author, lies in:

- unobtrusive learning of a joint model of a multi-user environment built up from a history of choices made by the users together and separately, where the term "history" denotes a time-stamped TV zapping log augmented with contextual information;
- the use of detectable parameters of social context as input features for adaptation to indefinable high-level social situations; and
- the use of an ensemble of diverse classifiers trained on target context data, in order to achieve more accurate adaptation to significantly different contexts.

The most interesting results achieved in this test case include firstly the exploration of simple implicit indicators of users' interests (such as the duration of programme's viewing and channel switching) for both individuals and families, the results of which show that such indicators exist but are somewhat less reliable for families. Secondly, it was demonstrated that a conventional approach to the modelling of group behaviour (acting upon individual preferences of group members) is not very suitable for use with families, as the choices made by individuals in many families significantly differ from the group choices because the satisfaction felt by individuals in groups is often influenced by altruistic feelings. These results suggest that learning a joint family model is a more appropriate approach than combining the individual preferences of family members. Thirdly, our findings demonstrated the feasibility of using implicit interaction data only, since the resulting accuracy of the proposed approach appeared to be comparable to that of recommender systems for individuals, requiring explicit interaction efforts, despite the fact that group modelling is generally more difficult. Last but not least, this study suggested new diversity measures for context-adaptive ensembles and demonstrated the feasibility of employing ensembles in cases where diverse classifiers can be built without any notable increase in training or fusion times, in that a lightweight ensemble of just four members allowed for more accurate adaptation in the tests than did any single classifier alone.

Regarding the main drawbacks of conventional context adaptation approaches (reliance on domain knowledge and significant explicit interaction efforts), these were addressed in several ways. Dependence on domain knowledge was reduced by:

- defining only context types (social and time contexts) and reading exact sets of context cues (i.e. the members of each family) from the data, e.g. one family may have consisted of two adults and two children and another of one adult and three children; and
- learning family practices from data instead of enforcing practices chosen by application designers.

Explicit interaction efforts were made optional. The decision to employ a diverse set of base classifiers in which the same low-level models are used in half of these base classifiers and lazy reasoning methods in the other half allowed a further reduction in data collection time. Reasonably accurate recommendations were provided after 3.5 weeks of data collection, which is not a long time for such a lifelong activity as TV viewing, moreover that adaptation to new users is a well-known problem also in systems, relying on explicit user effort.

3.5 Case 3: affect recognition

As noted in the literature review, research into context-adaptive emotion recognition is just starting, and both context-adaptive and context-independent systems are usually trained on fairly large quantities of manually annotated data. This means that existing works were not aiming at user-controllable adaptation. Furthermore, the majority of works on the adaptation of affect recognition systems reviewed here had collected their data in controlled environments, where the ways of expressing the same emotion in different contexts were fairly similar.

By contrast, our work studied adaptation to contexts in which the ways of expressing the same emotion differed considerably, and the data were collected in uncontrolled environments, but despite these challenges we were still aiming at allowing end users to control adaptation. In order to achieve this goal, a GUI for quick data annotation was designed, and a method for using small sets of labelled data for adaptation purposes was proposed.

The training of a model for a new context, either from scratch or by transferring knowledge from other contexts based on a small amount of explicit user feedback is described in Publication III.

3.5.1 Application scenario

The adaptation of an affect recognition system to situational changes was studied in an application designed for detecting whether the audience at a show is excited by the show or not [Publication III]. Events of three types in which ways of expressing excitement differ notably one from another were considered as contexts: a concert, a circus performance and a sports match.

Audience excitement recognition may help in detecting the highlights in a show [Joho 2011], but due to large variety of shows, system developers cannot fully train all the models beforehand. Thus our work assumed that system developers should train audio and video analysis models, and also a fusion module to some extent. Then, when a new context emerges, the end user should be able to adapt the fusion module of the emotion recognition classifier by annotating selected moments in on-going events, possibly while participating in these events or watching them on TV (see Figure 9). The parameters of the fusion module should be updated immediately after acquiring a certain relatively small quantity of user-labelled data, so that the highlights can be further classified in event-specific way. The resulting fusion model can be stored along with a user-provided context name or automatically acquired contextual features (e.g., the title of the TV programme or the GPS coordinates of a circus) and can be retrieved the next time that the same situation occurs.

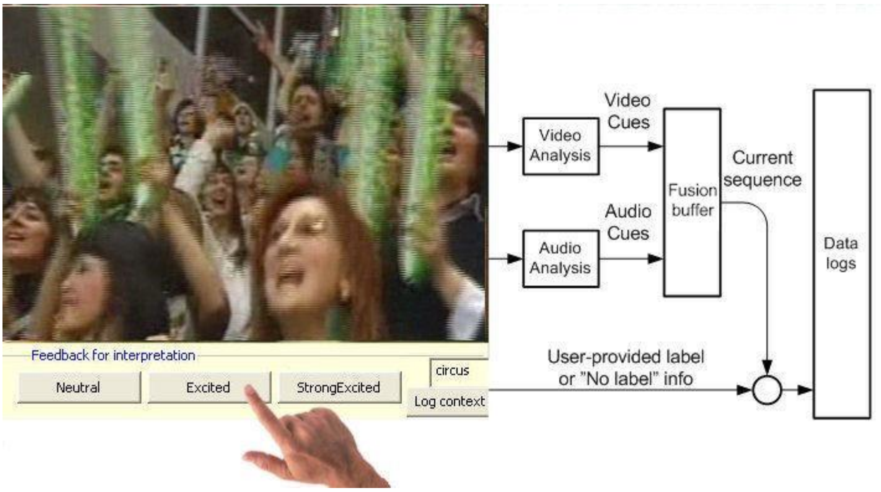


Figure 9: Annotation of audience reactions simultaneously with video viewing [Publication III].

In our work the cues for input to the multimodal fusion module were produced by context-independent audio and video analysis modules aimed at the recognition of certain data classes such as whistling, speech, motion (of objects or hands around human faces), etc. This scenario has the following specifics:

- *Context factors to adapt to*: social, environmental and historical factors;
- *Adaptation time*: a few minutes, otherwise users may tire;
- *Permitted degree of user control*: any;
- *Expectations regarding changes in input cues*: the context may affect the accuracy of recognising certain behavioural classes (e.g. changes in background noise always affect the accuracy of audio classification); data availability may depend on the situation (e.g. if a camera is positioned some distance from the audience or points in the other direction, video analysis may provide no useful results); the meaning of the data may

depend on the situation (e.g. whistling may come from the audience or from the show itself);

- *Costs of data acquisition vs. data quality*: implicit feedback is not acquired naturally; acquisition of explicit feedback is feasible because labelling takes only a few minutes and helps to improve the classification results for the next hour or more; and
- *Variability of situations*: many diverse and fairly indefinable situations can occur: although event types can be described, customary behaviour in these events cannot be reliably pre-defined. Audio and video backgrounds cannot be pre-defined either because they may vary dramatically between events of the same type.

3.5.2 Classifier design

The following single modality components were used: 1) a video analysis module detecting the optical flow and faces if they are not too small or significantly rotated; and 2) an audio analysis module trained on a database of clean audio data to recognise several audio classes: laughter, speech, applause, noise, silence, whistling etc. Thus audio and video analysis was context-independent, and the multimodal fusion models were context-adaptive. The following design choices were made:

- *Interaction type*: explicit, as implicit feedback is unavailable and not guaranteed to recognise changes in the interpretation of input cues,
- *Adaptation type*: model selection (as contexts may differ significantly); the models use historical context as a feature,
- *Data usage type*: 1) model-level knowledge transfer, 2) no data on initial contexts,
- *Runtime training*: custom,
- *Runtime training supervision*: partially supervised, as explicit feedback is never abundant (conventional unsupervised training for HMM and conventional supervised training for SVM were also tested), and
- *Reasoning methods*: fusion of audio and video cues by HMM employing MPM (maximum posterior marginal) decisions. In addition, fusion with HMM employing MAP (maximum a posteriori) decisions and fusion with SVM were tested.

The classifier design is presented in Figure 10. First stage: audio and video analysis components recognise selected behavioural cues, e.g. applause. Second stage: recognition of audience excitement via fusion of the cues. Fusion adaptation is performed by adjusting the probabilities of observing the behavioural cues in different hidden states of the model by means of explicitly annotated data. The probabilities are adjusted with a differential evolution algorithm. This design allowed rapid adaptation to any number of newly encountered situations under sparse user supervision, i.e. the user needed to provide a certain number of annotated samples and the fusion models were then immediately re-trained. Adaptation in this test case was therefore lightweight, because it required little in the way of user effort or computational resources.

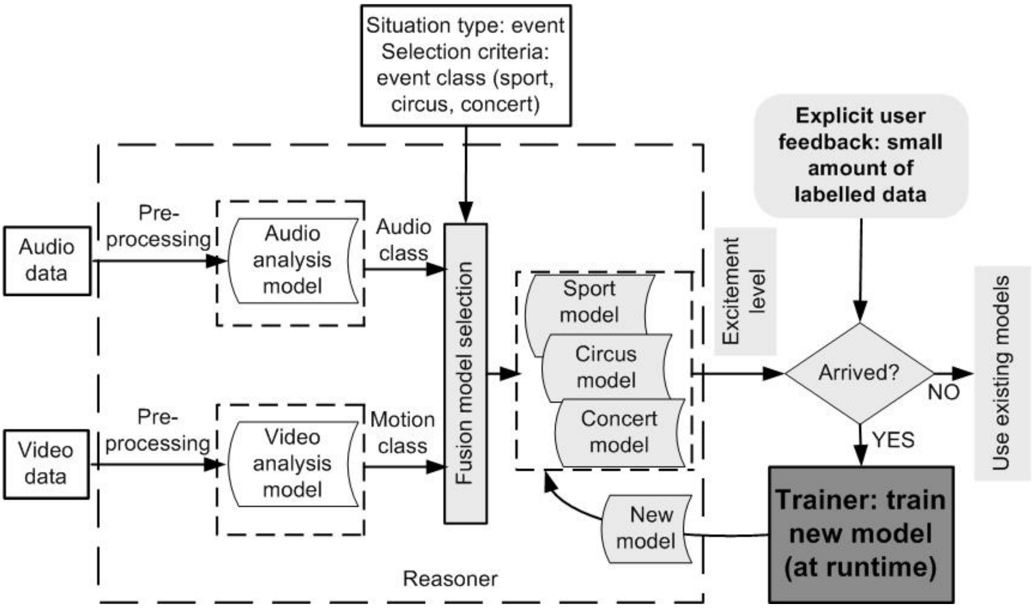


Figure 10: Adaptation for affect recognition: grey colour indicates components developed for multimodal fusion.

The following adaptation approaches were compared in the experiments:

1. *Context-specific models*:
   a. partially supervised context-specific HMM/cascaded HMM training – first a model is trained in an unsupervised way, applying the Baum-Welch algorithm to unlabelled data for the target context, and then improved using labelled data for the target context;
   b. conventional context-specific HMM: an HMM model trained in an unsupervised way, applying the Baum-Welch algorithm to the data for the target context;
   c. conventional context-specific SVM: fully supervised SVM trained on labelled data for the target context; and
2. *Model-level knowledge transfer*: a context-specific HMM model for the initial context (alternatively, a generic HMM model, i.e. a model trained on mixed data for all contexts) is adapted to the target context using labelled data for the target context.

To prove that context adaptation is indeed needed for the selected test case, these approaches were also compared with three generic models:

- Generic conventional HMM: an HMM model trained in an unsupervised way, applying the Baum-Welch algorithm to the data for all contexts;
- Generic partially supervised HMM: an HMM model trained in an unsupervised way, applying the Baum-Welch algorithm to the data for all contexts and further adapted using labelled data for all contexts; and
- Generic SVM: fully supervised SVM trained on labelled data for all contexts.

3.5.3 Experimental results

Altogether three hours of data on various sports events, circus shows and storytelling events were used for the experiments (one hour per situation). Ten minutes of data per situation was used for training, and the rest for testing. The HMM was trained to recognise three states of the audience: neutral, excited and very excited, these labels being acquired simultaneously by video viewing (as shown in Figure 9), and an annotator was free to decide when to provide labels. Labels were therefore obtained mainly for unambiguous data samples, e.g., if the audience's applause overlapped with music, the annotator labelled the moments when the applause was clearly recognisable as ''excited'' or ''very excited'', but not the moments when the music dominated over the applause. Labelling and training took altogether approximately 10 minutes per context, and during this time only 100–150 labelled training samples per context were acquired.

The comparison of several adaptation approaches confirmed the feasibility of employing MPM decisions and modifying the model parameters with a differential evolution algorithm. This approach enabled very lightweight and fast statistical learning on small databases containing noisy training data.

The HMM models obtained via *model-level knowledge transfer* achieved an 80.9±1% average recognition accuracy (at the 95% confidence level) despite the presence of erroneous annotations: annotators may react slowly when emotions change rapidly and may label a previous emotional state instead of the current one. (More details concerning labelling errors and effect of training data quality are provided in Publication III.) The *context-specific HMM models* obtained via partially supervised cascaded training achieved a slightly higher average accuracy of 83.7±1%, but the conventional context-specific models achieved lower accuracies, 77.4±1% in the case of SVM and 74.3±1% for HMM. The generic models were even less accurate, SVM achieving 68.1±1% accuracy and HMM 69.2±1%. Due to the MPM decision rule, HMM adaptation remained robust even when as little as 25 annotated samples per situation were used, as in this case the re-adapted models achieved an average accuracy of 79.5±1%. Neutral and non-neutral reactions were distinguished with 77.3±1.5% average accuracy, whereas conventional context-specific HMM achieved only 66.1±1.5% accuracy when distinguishing between neutral and non-neutral reactions and context-specific SVM only 63.8±1% accuracy, because a dataset of 25 samples is too small for fully supervised SVM training. Unfortunately, these results could not be compared with those quoted in other studies, because the latter used different data, mainly data collected in controlled environments.

3.5.4 Discussion

In the affect recognition domain we studied rapid adaptation to fairly indefinable situations via runtime training performed immediately after the new situation has emerged and a small amount of labelled data has been provided by the end users. This approach allowed us to give the users control over adaptation without requiring significant efforts, i.e. the annotators themselves chose which samples to label, and labelled only fairly unambiguous ones (recognition of only unambiguous excitement should suffice for the detection of show highlights). For the partially supervised cascaded training of HMM as little as 25 labelled samples per context were sufficient for fairly accurate adaptation, whereas fully supervised SVM models achieved similar accuracies only when 100–150 labelled samples per context were used in training. Supervised training should therefore not be the first choice in cases of explicit interaction if unlabelled data are available. Conventional unsupervised HMM training resulted in significantly lower accuracy. Hence unsupervised context adaptation cannot be recommended for cases when input cues may change meanings.

Cascaded training has mainly been employed to date in deep neural networks [Bengio 2009]. The conventional way of training not-so-deep architectures is to use all the available data at once, whereas results presented in Publication III suggest that cascaded training achieves an increase in classification accuracy and a decrease in labelling effort also in not-so-deep architectures. Another uncommon design choice was to employ MPM decisions in HMM instead of MAP decisions. MAP decisions were also tested, but their adaptation was far less accurate. This result suggests that lightweight adaptation may benefit from not-so-conventional inference approaches, too.

Other outcomes were the following: firstly, modification of the observational probabilities in HMM allows changes in data availability and meaning to be handled very conveniently. Several preliminary tests were needed to find a reasonably accurate way of modelling the same data with SVM, and the resulting solution required the full re-training of two models for fusing audio and video cues for each situation. Secondly, model-level knowledge transfer for HMM resulted in lower accuracy than partially supervised HMM training, but the difference was not a dramatic one: the latter achieved 83.7 ± 1% average accuracy, while the former - 80.9 ± 1% average accuracy. Thus model-level HMM adaptation is feasible when the storage of raw data is undesirable. Regarding the choice of the model to adapt, the adaptation of various initial context models to the target contexts resulted in fairly similar accuracies in the tests, probably due to the notable differences between all the contexts. The adaptation of generic models resulted in a slightly higher accuracy, but training a generic model requires the storing of raw data on all the contexts. In this case it is more feasible to train a context-specific HMM model on the target context data in a cascaded manner.

One survey of audio-visual information fusion [Shivappa 2010] states that, although it is not well known how humans understand the complex world, the consensus is that an integration of information at different levels of the semantic hierarchy is necessary in order to complete this task. Other works agree that hierarchical information processing is characteristic of humans and beneficial for computers [Bengio 2009]. Thus the proposed cascaded approach to recognising certain behavioural cues first and then interpreting the cues in a context-specific manner may well correspond to human thinking. This approach also allows a significant reduction in data collection efforts, since training an audio classifier, for example, to recognise applause and laughter accurately just in a circus, where these sounds overlap with each other and with speech, music, the screams of artists and animals, creaks in the mechanisms etc., would require collecting over twenty types of mixed sounds, quite a large number considering that other sounds, too, emerge in other contexts.

The approach proposed here considers several aspects of situational influence: firstly, HMM naturally takes into account the previous emotional state, and secondly, situation-adaptable interpretation by adjusting the observational probabilities allows different meanings to be assigned to the cues in different contexts and permits the easy handling of missing data. In circus and concert contexts, for example, applause and laughter are the most useful sounds for detecting audience excitement, whereas in a sports context laughter is a sign of audience disappointment rather than approval. On the other hand, “whistling” is a sign of audience excitement in a sports context, but in a circus it is mainly the clowns who whistle, and in a concert context whistling is a very rare event. Visual cues can be missing in all contexts, too, e.g. when cameras point at a show. Last but not least, the proposed approach also allowed us to cope with inaccurate recognition of certain cues in different contexts, e.g. if ‘‘laughter’’ were to be frequently misclassified as a cue of some other kind because of a challenging audio background, the system would learn that the probabilities of observing laughter are more or less equal for all HMM states in this context and would rely on other cues instead.

In the approach proposed here the fusion model parameters are modified for each situation, and thus this approach is well suited for both fairly distinct and fairly similar situations, and for both easy-to-define and indefinable situations. The proposed lightweight adaptation approach could also be employed for partially supervised HMM adaptation for other purposes, especially since MPM decisions entail the same time complexity as conventional MAP ones.

3.5.5 Summary

As shown in the literature review, one of the novelties of the author's work lies in its goal: to allow end users to make rapid adaptations of the system for detecting show highlights to context, e.g. type of show, performers, TV channel, audio background etc. and to their own perceptions (the majority of works on context adaptation to date do not consider user-controlled adaptation). The main novelty of the solution, proposed by the author, lies in:

- detecting fairly high-level behavioural cues and interpreting them in a context-specific manner by modifying their observational probabilities;
- employing a differential evolution algorithm for model-level knowledge transfer of HMM; and
- employing assumption-free partially supervised cascaded training of the HMM.

The most interesting results achieved in this test case include, firstly, the comparison between MPM decisions and conventional MAP decisions, in which MPM decisions allowed more accurate adaptation because they used a more consistent search space. Secondly, partially supervised HMM training appeared to be notably more robust for reducing the size of the labelled dataset in this test case than did the fully supervised training of SVM. Thirdly, this test case demonstrated that use of classifier confidence is less beneficial in adaptation to notably different contexts than it is in conventional systems, and that the rejection of low-confidence fusion results did not significantly increase the classification accuracy as was expected, because several mistakes of low-level models in a row resulted in fairly high confidence in the erroneous emotion recognition results.

Regarding the main drawbacks of conventional context adaptation approaches (reliance on domain knowledge and the need for a significant explicit interaction effort), the dependence on domain knowledge was reduced by employing fairly generic audio and visual cues and assumption-free learning of their interpretations from the data (conventional semi-supervised training methods often employ certain domain-dependent assumptions). Dependence on explicit interaction efforts was reduced by: 1) allowing users to choose the samples to annotate; 2) modifying only the observational probabilities of HMM models; 3) proposing a method for learning from small sets of labelled data; and 4) employing unlabelled data whenever feasible. In the tests 10 minutes of data acquisition per context allowed fairly accurate adaptation even when just 25 labelled samples per context were obtained.

3.6 Case 4: user interaction

As seen in the review of the literature, little work has been done on the adaptation of user interfaces to social context and on learning adaptation models. By contrast, we proposed an approach to learning the dependence of interface preferences on various contexts, including the social context. The transfer of knowledge from other contexts in cases when only very short interaction histories of several users are available was studied in Publication IV.

3.6.1 Application scenario

Although the convenience of an interaction often depends on its context, manual interaction customisation would require too much user effort in the future, given that users would be interacting with numerous applications through different interfaces, e.g. a smart car, a smart shopping assistant etc. Support for group interaction poses additional challenges, in that humans tend to respect the preferences of their friends and family members, so that the preferred interface settings may depend on all the group members. It was suggested in Publication IV that the effort involved in manual customisation could be reduced by predicting the interface preferences of individuals and groups for new (unseen) combinations of applications, tasks and devices. UI adaptation was studied in the scenario of a new user or new group of users (referred to as the target user/group) launching a new application or task in such a manner that the application interface is automatically adapted to that target user/group's predicted preferences. Hence the target context in this case could be a new application, a new task or a new user group. The customisation choices of different users/groups regarding certain UI elements then served as input cues to the fusion module. This scenario has the following specifics:

- *Context factors to adapt to*: social, task and computational factors;
- *Adaptation time*: practically instant, because users want to use an application immediately after its launching;
- *Permitted degree of user control*: any;
- *Expectations regarding changes in input cues*: the influence of certain interface features on user convenience changes according to situation;
- *Costs of data acquisition vs. data quality*: both feedback types are acquired naturally. When users change the interface settings manually they provide explicit feedback regarding the quality of the predictions, and when they start using the interface with the predicted settings without changing them they provide implicit feedback; and
- *Variability of situations*: many distinct situations, both describable (e.g., screen sizes) and indefinable (social rules operating in groups).

3.6.2 Classifier design

The users' choices regarding various interface features served as input cues. (An "interface feature" is any aspect of the interaction, e.g. sensor-based activity recognition; type of information presented, e.g. a tool tip, reminder etc.; input/output modality, such as GUI or audio etc.) To reduce the interaction effort required of each user, we employed data on the user community and assumed that the community size was fairly small because the test subjects in our user study agreed to share their interaction preferences with their friends but not with the whole world. The following design choices were made:

- *Interaction type*: explicit and implicit feedback, as both are naturally acquired when users customise applications or refrain from doing so;
- *Adaptation type*: a selection-based ensemble of knowledge transfer strategies, because 1) small databases do not allow the use of sophisticated learning methods, 2) user communities can provide sufficient data for comparing the accuracies of ensemble members, and 3) this comparison is not computationally expensive due to the small size of the database;
- *Data usage type*: raw data, as the number of persons sharing their interaction preferences with each other is not so large as to make storage of these preferences problematic;
- *Runtime training*: none, due to the very short interaction histories;
- *Runtime training supervision*: the accuracies of the ensemble members are compared when operating with the available data; and
- *Reasoning methods*: kNN, majority voting and certain heuristic strategies.

These choices allow rapid adaptation to newly encountered situations, provided that different users use the same names for the emerging situations or ontology is used for finding synonyms. The chosen adaptation type belongs to a "*mixed data ensemble*" class. The classifier ensemble was built up as illustrated in Figure 11. First, situation-dependent interaction preferences were collected for different individuals and groups with respect to the various applications, tasks and devices, and these data were used for evaluating the accuracies of all the ensemble members with respect to all the interface features. Then, when a target user/group enters a situation not previously encountered, the user/group preferences for each interface feature are predicted by the ensemble member that had achieved the highest prediction accuracy for this feature in the past.

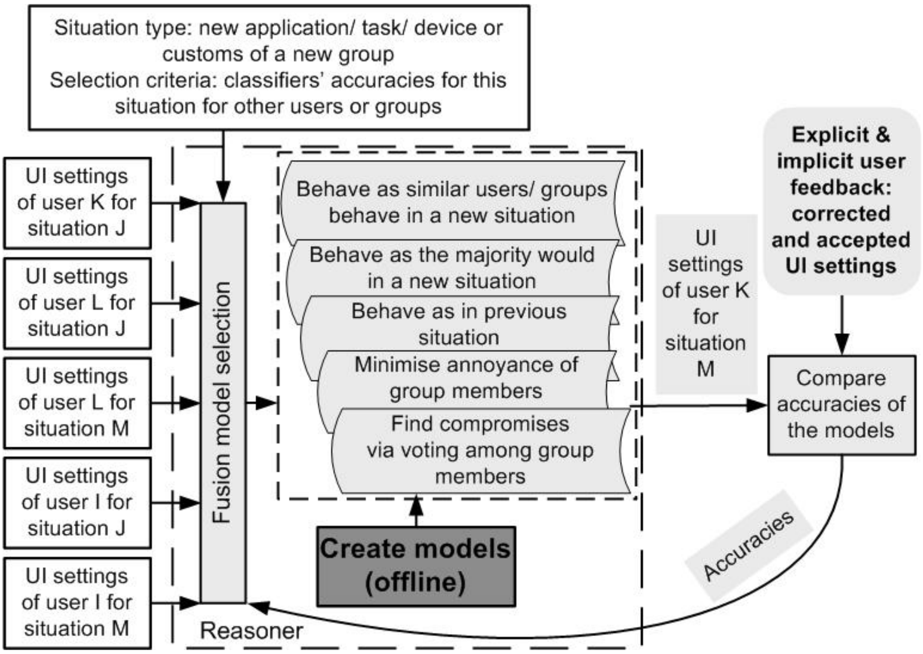

Situation type: new application/ task/ device or customs of a new group
Selection criteria: classifiers' accuracies for this situation for other users or groups
UI settings of user K for situation J
UI settings of user L for situation J
UI settings of user L for situation M
UI settings of user I for situation J
UI settings of user I for situation M
Fusion model selection
Reasoner
Behave as similar users/ groups behave in a new situation
Behave as the majority would in a new situation
Behave as in previous situation
Minimise annoyance of group members
Find compromises via voting among group members
Create models (offline)
UI settings of user K for situation M
Explicit & implicit user feedback: corrected and accepted UI settings
Compare accuracies of the models
Accuracies

Figure 11: UI adaptation: grey colour indicates components developed for multimodal fusion.

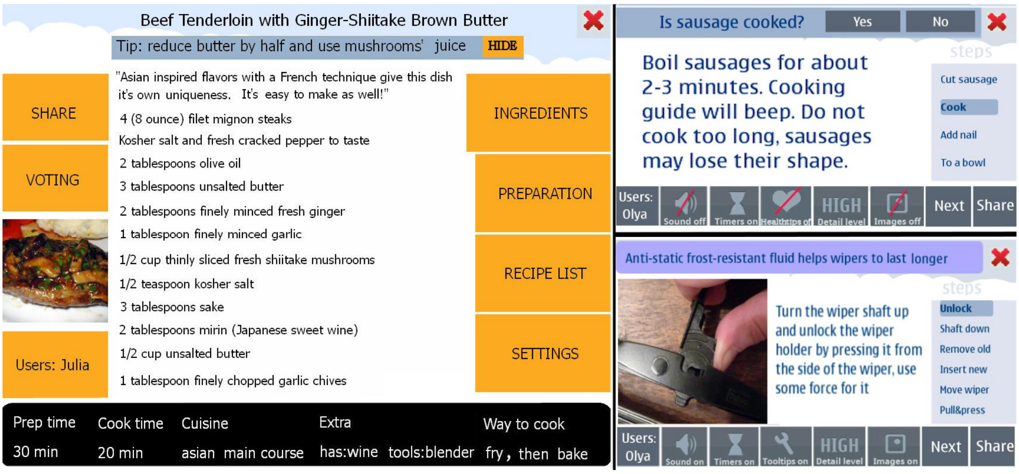
The conventional way of building ensembles is to employ pattern recognition methods as the base classifiers, i.e. mappings between the data and the classification result. It was proposed in Publication IV that an ensemble of knowledge transfer strategies should be employed for enabling very quick adaptation, and that the base classifiers in this ensemble should model different ways of mapping data for one context onto data for another context, namely certain typical forms of behaviour exhibited by humans when entering a situation they have never encountered before. For example, an individual may behave in the same way in a new situation as in some other situation, or in the same way as the majority of other people would behave in this situation, or in the same way as those other users who behaved similarly to the target individual in some other situation. Likewise, a group may behave in the same way as other groups, but may make their choices by voting among the group members. Thus humans may transfer knowledge about previous situations in various ways. Behaving in a new situation in the same way as in another one denotes full transfer of knowledge, while behaving in a new situation in the same way as others would behave in this situation means that old knowledge is ignored, and behaving in the same way as similar users denotes a transfer of similarity. The kind of behaviour a user may exhibit depends on the specifics of the target context and on the similarity between initial and target contexts.

"Behaving in a new situation in the same way as in a previous one" (for example, choosing same interface layout in both situations) is a suitable strategy when the initial and target contexts are similar; "behaving as the majority of others would in this situation" is a suitable strategy for target contexts that seriously affect users' preferences: for example, in the study with the recipe recommender system nearly all test subjects were interested in a "way to cook" (e.g., baking or frying) when they looked for recipes for "cooking outdoors" context. Similarity transfer could work for all context transitions if users were to maintain their similarity across all contexts, but this may not be the case. Early technology adopters, for example, may accept uncommon interaction features in new applications more easily than would conservative-minded people, whereas similar attitudes towards a healthy lifestyle may lead to similarity in interests regarding food and nutrition, but similar attitudes towards technology adoption do not necessarily imply similar attitudes towards a healthy lifestyle. Accordingly, before similarity transfer can take place it is necessary to check whether similarity can indeed be transferred from the initial context to the target one. User community data were employed in Publication IV for checking this. The automatic selection of the most appropriate ensemble member for each target user/group was based on a comparison of the prediction accuracies of all the members for all non-target users/groups whose preferences for the target situation had already been ascertained. Consequently, adaptation was very lightweight in this test case because it required very little user effort and almost no computation.

3.6.3 Experimental results

The proposed approach was tested using interaction preferences acquired from 21 subjects for three applications, a cooking assistant, a car servicing assistant and a recipe recommender, and the preferences of another 23 persons for the recipe recommender. Examples of customised interfaces for these applications are presented in Figure 12.

The preferences were predicted for diverse interaction features, including audio input and output, sensor-based activity recognition, reminders, health and tool tips, and shop offers. The results show that the proposed approach is suitable for cases where no long interaction histories are yet available, and that it is not restricted to similar interfaces, screen sizes or application domains. Again the ensemble achieved a higher average accuracy in the tests than any of its members alone: 72±1%. User studies employing test subjects from two user communities [Publication S4, Publication IV] demonstrated that adaptation to social rules is considered very important for assistive applications and that lightweight adaptation solutions proposed are generally well accepted. It is impossible to compare these results with the state of the art because the prediction of interaction preferences is a novel problem, but a comparison with recommender systems shows that a success rate of 70–75% is usually considered a good result [see, for example, Adomavicius 2007, Blanco-Fernandez 2010].



Beef Tenderloin with Ginger-Shiitake Brown Butter

Tip: reduce butter by half and use mushrooms' juice

"Asian inspired flavors with a French technique give this dish it's own uniqueness. It's easy to make as well!"

4 (8 ounce) filet mignon steaks
Kosher salt and fresh cracked pepper to taste
2 tablespoons olive oil
3 tablespoons unsalted butter
2 tablespoons finely minced fresh ginger
1 tablespoon finely minced garlic
1/2 cup thinly sliced fresh shiitake mushrooms
1/2 teaspoon kosher salt
3 tablespoons sake
2 tablespoons mirin (Japanese sweet wine)
1/2 cup unsalted butter
1 tablespoon finely chopped garlic chives

Users: Julia

Prep time	Cook time	Cuisine	Extra	Way to cook
30 min	20 min	asian main course	has:wine tools:blender	fry, then bake

Is sausage cooked?

Boil sausages for about 2-3 minutes. Cooking guide will beep. Do not cook too long, sausages may lose their shape.

Users: Olya

Sound off, Timers on, Healthtips off, HIGH Detail level, Images off

Anti-static frost-resistant fluid helps wipers to last longer

Turn the wiper shaft up and unlock the wiper holder by pressing it from the side of the wiper, use some force for it

Users: Olya

Sound on, Timers on, Tooltips on, HIGH Detail level, Images on

Figure 12: Left – laptop GUI of the recipe recommender, top right – phone GUI of the cooking assistant, bottom right – phone GUI of the car assistant

3.6.4 Discussion

The main goal of our work was to solve the cold start adaptation problem, i.e. to predict interface preferences for contexts not encountered earlier by the target user/group, and to do so without a long process of data collection. Thus very small datasets were used for the predictions and the choice of reasoning methods was very limited. The proposed ensemble allows adaptation to both easy-to-define (e.g., "phone" vs. "laptop" device) and indefinable situations (e.g., group practices), provided that it is known for which situation (e.g., "alone" vs. "in a group") the data for the various users/groups were acquired. This approach does not require describing social rules for each situation but only the defining of transfer strategies. It is suitable both for cases where the initial and target contexts do not differ significantly and for cases where they do differ, and it does not require any domain knowledge for defining context similarity. When predicting interface preferences for a car servicing task, for example, another car servicing task would be a fairly similar initial context, and a significantly different initial context would be the task of finding a recipe for a party, for instance. The proposed approach handled both cases fairly accurately without any assumptions regarding context similarity and without relying on large databases of interface preferences.

The tests also demonstrated that it is feasible to transfer user similarities across some contexts, but not all, and that even when like-minded persons remain similar to each other in different contexts, their degree of similarity changes. Two ways of transferring similarity were compared in the tests: conventional kNN, which does not weight the opinions of like-minded persons by their distance from the target users, and conventional CF, which performs such weighting. In the tests the kNN method outperformed CF, especially in cases where the initial and target contexts differed significantly.

The proposed ensemble was not compared in Publication IV with any of the other fairly popular approaches to the transfer of knowledge across contexts, but our earlier work [Publication S5] tested two other approaches using data for the same 21 subjects collected for the cooking and car assistant applications, which have very similar interfaces but work in significantly different domains. The first of the knowledge transfer approaches to be tested was a fairly simple one: to add or subtract a shift vector from the vectors of the preferences for each user (or group of users), similarly to [Baltrunas 2012]. This approach is based on the assumption that different interaction features become more or less useful in different contexts, and that their usability gains (or losses) depend on the features and contexts, but not on the users. The average accuracy of this approach appeared to be lower than that of kNN, despite the fact that kNN failed for some context transitions. Another approach that was tested was one of the most common ones: to adapt the similarity measure. For adaptation we used a differential evolution algorithm aimed at minimising the prediction error for all non-target users. This approach was indeed successful, especially in cases where the users' preferences were fairly similar in the initial context but differed significantly in the target context. The average accuracy of this approach was nevertheless fairly similar to that of the ensemble of knowledge transfer strategies, while the computational time was notably longer.

Thus employing an ensemble of computationally inexpensive transfer strategies is a more feasible approach, provided that 1) the users in the community do not significantly differ in their culture, 2) the ensemble members cover a sufficient range of human knowledge transfer strategies, and 3) the dataset is adequately large for determining the most accurate methods for each context transition. The first requirement was satisfied here, as, although the test subjects had fairly different attitudes towards the technologies concerned, their attitudes towards the relevant social rules were fairly similar, e.g. they all valued group communication more highly than cooking efficiency, and they all respected each other's diet preferences. The second requirement was also satisfied here, but in general the choice of adequate knowledge transfer strategies is something that requires thorough consideration. "False mirror" behaviour, for example, was not considered in this study, and the subjects clearly did not aim at making the initial and target interfaces as different from each other as possible. Nevertheless, such behaviour might be a workable knowledge transfer strategy in some other applications, e.g. in the e-learning domain a student may want to diversify exercises. Regarding the third requirement, selection of the best ensemble members was not robust to small variations in users' behaviour in the tests unless the data size exceeded the ensemble size by at least a factor of three.

3.6.5 Summary

Since the author's work was aimed at developing a method for adapting the interfaces of personal applications to users and contexts without requiring any appreciable manual customisation efforts, the author developed a method for predicting interface preferences for newly encountered contexts, i.e. applications, tasks, screens and newly assembled user groups. The main novelty of the solution lies in:

- employing a selection-based ensemble of knowledge transfer strategies in which some strategies are suitable for fairly similar contexts and others for fairly different contexts; and
- employing data for several contexts for learning, which knowledge transfer strategies are better suited to particular context transitions.

The most interesting results achieved in this test case include firstly the selection of knowledge transfer strategies that are suitable for transitions between both similar and dissimilar contexts and for both individuals and groups. Secondly, the test results demonstrated that the degree of user similarity changed between contexts and that users were more similar in their choices of knowledge transfer strategies than in their choices of interface settings. Thirdly, comparison of the proposed ensemble of fairly simple knowledge transfer strategies with a conventional approach to adapting the user similarity measure to contexts demonstrated that both methods achieved fairly similar levels of accuracy, but that ensemble-based adaptation was notably quicker. Last but not least, the results show that it is very important as far as user acceptance is concerned to respect the relevant social rules and that the use of data on user communities can facilitate the obtaining of the users' trust, in that users may be more willing to rely on the opinions of their acquaintances than on the choices made by application designers. On the other hand, the proposed ensemble is likely to fail if the members of a user community do not share fairly similar attitudes towards the social rules.

Regarding the main drawbacks in the conventional context adaptation approaches (reliance on domain knowledge and significant explicit interaction efforts), the dependence on domain knowledge was reduced by 1) defining data types (UI elements that can be customised independently of other elements vs. UI elements that are customised as a group, e.g. by selecting N options from a larger set) and reading exact sets of data features (e.g. audio reminders, tool tips etc.) from the data, and 2) including in the ensemble both members that are suitable for transitions between similar contexts and members that are suitable for transitions between significantly different contexts, and selecting the best members by comparing their performances on the available data. The dependence on explicit interaction efforts was reduced by 1) employing whatever data the users provided (usually a mixture of implicit and explicit data), 2) employing data for user communities, and 3) using very short interaction histories (predictions of the preferences of target users in a new context require knowledge regarding their preferences for just one initial context) and simple reasoning methods.

4. Conclusions and recommendations

The present work was focused on the situational adaptation of multimodal fusion models by utilising explicitly or implicitly acquired runtime interaction results, and on finding ways of reducing the explicit interaction effort. Adaptation approaches were suggested for class-level fusion, as adaptation of feature-level fusion is usually more computationally and data-demanding. For the same reason, no adaptation of the components for the analysis of individual modalities was performed, but instead the same single modality models were used in all situations. Accordingly, the proposed lightweight adaptation approaches are not suitable for context transitions in which the analysis of single modalities fails, nor do they provide for arbitrarily evolving feature spaces, as at least the types of data and context features should be defined at the design stage. (Exact sets of features of each type can be read from the data at the runtime stage if the classifiers employed allow one to deal with new features. New features are easily handled by similarity-based methods, for example, and by methods for learning a model for a target situation from scratch.)

Despite the limitations, lightweight adaptation is suitable for dealing with various practical problems, especially in cases when an alternative full-blown adaptation cannot be performed at all or would require substantial data collection effort. We did not, for example, find any reasonable alternative to lightweight adaptation in the two test cases, user interface adaptation and affect recognition. In this work the lightweight adaptation was based on data, implicitly acquired from the users in the course of using one application for one task, which typically took from 5 to 20 minutes. Longer interaction histories were not available for the cold start adaptation problem concerned here. Other works, suggested methods to use interaction data in interface adaptation, required notably longer data collection: for example, one study collected posts, written by the test subjects in social networks since their registration. Likewise, full-scale adaptation of the affect recognition system would have required a considerable amount of effort for collecting and annotating context-specific audio and video data and might have failed anyway because state-of-the-art audio processing methods do not allow the reliable recognition of as many classes of mixed sounds as were observed in the data, while state-of-the-art video processing methods do not allow the reliable detection of small, non-frontal faces. As end users are even less likely to invest their efforts in data collection than are application designers, it would not be feasible to offer them full-scale adaptation as the only option.

Depending on the domain specifics, we were able to suggest several context adaptation methods which require relatively short adaptation times (from a few seconds up to 10 minutes, depending on the task at hand) and allow for reducing the need for domain knowledge and explicit interaction efforts. In particular, we did not use detailed domain knowledge-based assumptions regarding influence of context on user and system behaviour, frequently employed in conventional adaptation approaches (for example, that in one context a show audience expresses excitement by screaming, whereas in another context it is not allowed to scream at all).

The first proposed approach is suitable only for adaptation to contexts that can be pre-defined at the design stage, as it aims at reducing the explicit interaction effort required for inference.

Other approaches allow adaptation to situations that emerge during the runtime and aim at reducing the explicit interaction effort required for model training. Two of these approaches perform knowledge transfer between contexts, and one approach trains models using the target context data. Nevertheless, all these approaches are capable of handling both cases of fairly similar contexts and cases of rather different ones without relying on any domain knowledge-based assumptions. Instead, the ways of handling newly emerging contexts are learned from the data.

The adaptation methods proposed for the present tasks might well be used in a broader range of applications. *Cascaded inference* is a method for providing conclusions based on implicit data and on requesting explicit interaction only if this attempt fails. In biometric verification tests this approach allowed a significant reduction in explicit verification effort compared with the parallel fusion architecture (used in the majority of the existing biometric systems) at the cost of only slightly lower accuracy, despite fairly high error rates in components classifying implicitly acquired data samples. Methods for building *cascaded systems,* proposed in this work, may therefore be worth considering in various applications where classification can be based on any combination of implicitly or explicitly acquired data samples, provided that the application-dependent requirements regarding overall classification accuracy are satisfied. Emotion recognition systems, for example, could first use unobtrusively acquired voice and face data and then prompt the user to look at the camera or answer a question. In the absence of any security restrictions, inference models could be trained during the run time if users were to agree to label their emotional states.

Model-level knowledge transfer for learning context-specific interpretations of input cues is a novel method for adapting graphical models that employ fairly generic high-level cues with context-dependent meanings. (Up to date, just a few other model-level knowledge transfer methods were proposed, and mainly for the classifiers employing fairly low-level input cues [Caridakis 2008, Yang 2009, Zhang 2005].) The hierarchical reasoning approach, employed in this work, may well correspond to human thinking as there is evidence of hierarchically organised reasoning in humans [Bengio 2009]. This approach may be used for modelling not only the emotions expressed by crowds, but also the emotions of individuals. It can be used also for activity recognition as various human activities are often modelled with HMM.

Our findings also suggest that a differential evolution algorithm aimed at minimising errors in a small set of labelled target context data should be employed for modifying the parameters of the initial models. This approach is one of the most lightweight approaches, proposed to date: it requires neither large datasets (usually only selected model parameters are modified) nor long computational time. The differential evolution algorithm is fast, it does not require a differentiable penalty function and it does not easily get stuck on local minima. Consequently, it could possibly be employed for the situational adaptation of models created by other algorithms, too, e.g. training of neural networks using evolutionary algorithms has already been shown to be successful, although not yet for context adaptation purposes.

Selection-based ensembles of knowledge transfer strategies offer a novel method for rapidly adapting applications in which a limited set of strategies can be defined for mapping data applying to one context onto data for another context (where “data” may denote system inputs or outputs or both) and in which data on user communities are available for selecting the best strategy for each context transition. This approach can be more or less lightweight depending on the number of ensemble members and the extent to which these members are themselves lightweight. Methods for mapping system outputs, for example, are usually more lightweight than methods for mapping inputs. This approach can handle both similar and dissimilar contexts if different ensemble members are optimised for different degrees of similarity between contexts. To the best of our knowledge, up to date such ensembles did not employ methods to map system outputs in one context onto outputs for another context: the only other proposed ensemble of knowledge transfer strategies employed relevance feedback techniques as base classifiers [Yin 2005, Yin 2010].

It is proposed here that the best ensemble member should be selected for each output class separately, an approach which was found to be beneficial because users do indeed often use different forms of logic when making decisions regarding different output classes. Ensembles of knowledge transfer strategies are rarely, if ever, employed nowadays, but our results suggest that this approach deserves more attention. Ensembles of behavioural models (e.g., “same as in the previous situation”, “opposite to the previous situation”, “same as for persons who were similar to the target user in previous situation” etc.) may be useful in helping to choose the level of difficulty of learning tasks in e-learning systems, for example, while an ensemble of affect recognition models for “formal”, “moderately formal” and “informal” settings may be used for distinguishing between these types without explicitly defining the degree of formality. Ensembles of knowledge transfer strategies may also include sets of user or context similarity measures or sets of user feedback utilisation methods. Such ensembles could possibly be built up not only using data for user communities, but also using multimedia databases, for example (e.g. it may be worth trying knowledge transfer for selecting the most discriminative data cues).

Cascaded training is a method for training a model first in an unsupervised way, using unlabelled data, and then improving on this model by means of a differential evolution algorithm, in which the parameters of the initial model are optimised using a small set of labelled data. This fairly lightweight approach is suitable for cases where both labelled and unlabelled target context data are available. In our work this approach was proposed for recognising audience excitement by means of an HMM (hidden Markov model) classifier employing MPM (maximum posterior marginal) decisions, but it can also be applied to the task of recognising emotions in individuals. Cascaded training is currently employed mainly for training deep neural networks [Bengio 2009], but not for HMM training. Our results suggest that the deep architectures are not the only ones that may benefit from such an approach. Our experiments showed that partially supervised cascaded training of HMM models employing MPM decisions allows more accurate classification than the corresponding partially supervised cascaded training of HMM models employing more conventional MAP (maximum a posteriori) decisions. As MPM decisions have the same time complexity as MAP ones, they could possibly be beneficial in the HMM-based modelling of various human activities.

One more approach, studied here, is *training of ensembles of diverse classifiers on target context data.* Employing diverse reasoning methods was shown to increase classification accuracy in conventional classifier ensembles, but we are aware of only one other work, employed diverse base classifiers in a context-adaptive system, aimed at minimising explicit user effort [Zhang 2009]. To the best of our knowledge, we were the first to use classifier ensemble in context-aware recommender systems and to suggest diversity criteria for its base classifiers.

This approach can be lightweight if the number of output classes is fairly small and the sets of labelled data are also small (for example, in image retrieval study in [Zhang 2009] the users provided only five positive examples, relevant to their query), but can otherwise require a considerable amount of computational time. This approach is suitable for applications in which it is more important to gain user trust than to reduce computational time, and where the acquisition of labelled data is not so tiring for users, e.g. when implicit interaction results are available or when many sets of negative examples can be selected from unlabelled data. This approach handles noisy interaction data better than a single classifier. Training an ensemble of diverse classifiers on target context data may be beneficial for adaptation to significantly different contexts in many application domains, e.g. affect recognition or multimedia analysis.

Adaptation solutions involving several classification algorithms (HMM, SVM, similarity-based methods etc.) were proposed in our test cases, and the analysis of adaptation experiences allowed important application characteristics to be identified and adaptation design guidelines to be suggested on the basis of heuristic classification of context transitions [Publication VI]. Heuristic measures are usually less reliable than numerical ones, but heuristic guidelines are appropriate for situational adaptation because application designers cannot predict all the contexts in which their applications might be used, and thus they would not be able to employ numerical measures in any case. We therefore proposed qualitative guidelines for adaptation design under different sets of application requirements, with the aim of helping to make high-level design decisions such as the choice of adaptation type (e.g., ensemble vs. model selection) and data usage type (e.g., whether to utilise or ignore knowledge regarding other contexts when learning ways of reasoning in the target context). These guidelines were developed after analysing the influence of various context types on the classification methods employed in several state-of-the-art application domains and in our own work. Hence the guidelines proposed here should be applicable to various applications in which adaptation mistakes are not likely to cause serious problems for users.

5. Summary

As has recently been observed, conventional data mining research is driven more by scientific interests (e.g. the objective to suggest innovative algorithms) than by practical considerations (e.g. the need to solve a certain real-world problem). At the same time data mining applications need to take into account the whole problem-solving process, which includes user interactions, the influence of environmental factors etc. [Cao 2010]. This observation also holds good for classification systems. It is for this reason that our work was aimed at finding practical solutions for the runtime situational adaptation of classifiers in cases where the use of conventional approaches would require too much effort from end users. Such practical adaptation solutions are referred to here as lightweight, because they allow a notable reduction in user effort without the need for any increased computational resources. The experimental results reported here in four application domains that differ significantly in their requirements, data availability and types of situational changes demonstrate that the proposed lightweight adaptation approaches achieved a notable increase in application convenience or classification accuracy.

Developing lightweight adaptation solutions required addressing the following drawbacks entailed in conventional classifiers: dependence on domain knowledge, the need for explicit interaction efforts and computational time. The proposed lightweight adaptation enables the reducing of dependence on domain knowledge firstly by employing context-independent input cues, e.g. by suggesting the use of fairly generic input cues that can be interpreted differently in different contexts (and also interpreting the absence of cues in context-specific ways). Our work also suggested the use of pre-defined context-independent types of cues and the reading context-specific sets of cues of each type from the data during the runtime. Secondly, our work reduced the dependence on domain knowledge by employing forms of data-driven learning that did not rely on any designer-provided assumptions, i.e. the learning of context-specific fusion models, and/or context-specific selection of the most appropriate fusion strategies, from the data. The need for explicit interaction and learning time was reduced by finding ways to learn from very small sets of explicitly acquired data and/or finding ways to make efficient use of implicitly acquired data, unlabelled data and data for user communities whenever available.

In all the application domains our work was aimed at solving problems that had not yet been addressed in current state-of-the-art research. In the biometrics domain the runtime training was not allowed for security reasons, so that all the models were trained beforehand. Hence when studying biometric verification, we aimed at reducing explicit user effort at the system inference stage, i.e. at making verification less obtrusive, a problem that is rarely addressed by current research in biometrics even though it is well known that users tend to ignore annoying security measures whenever possible [Wright 2008]. Our work demonstrated that even when runtime learning is not allowed, systems can be made more user-friendly if they are carefully designed.

For the other three test cases we proposed methods for learning reasoning models during the runtime without requiring any significant user effort. In the user interaction domain we attempted to predict interface preferences for contexts not previously encountered by the target users, e.g. for a new user group or a new application. The majority of current works on interface adaptation do not provide for multi-user environments and employ manually specified adaptation rules in other cases, whereas we were able to demonstrate the advantages of the learning of preferences. In the TV programme recommendations domain we aimed at increasing recommendation accuracy by taking into account the specifics of each family, whereas existing research is either concerned with recommendations for individuals or groups of friends or employs less adaptive solutions based on designer-provided logic. For affect recognition systems we proposed a method for fast user-controllable adaptation to new situations that required the end user to spend just 10 minutes for labelling data samples for each target context. Context-dependent emotion recognition is a fairly new research problem, and to the best of our knowledge, none of the works published on this problem has so far employed user feedback for increasing classification accuracy.

Research into lightweight adaptation is just beginning, and user acceptance of it is still an open question. We studied user acceptance only in the interface adaptation test case, where users' attitudes were largely positive. As in other studies of the user acceptance of adaptive applications, we observed that acceptance does not depend only on prediction accuracy. Many other factors play important role, e.g. user personality, screen size and the ability of the application to respect social rules. For example, some of our subjects did not accept 100% accurate predictions because they did not want their devices to be “too smart”, whereas others appreciated predictions of fairly low accuracy. Acceptance may also depend on the implementation of the reasoning. In our case the predictions were based on user community data, and many subjects said that they liked this because of the confidence they had in their acquaintances. This result suggests that the lightweight knowledge transfer method proposed here may not only allow a reduction in the need to collect data for the target situation, but may also facilitate users' trust. We also observed that in cases of group use the acceptance shown by the group members depended on the satisfaction felt by the other group members, in both a positive (i.e. users may give up their preferences more easily) and a negative way (i.e. users may be dissatisfied but unwilling to insist on their own wishes).

Regarding the willingness of end users to invest efforts in runtime system adaptation, numerous studies show that users may even provide detailed feedback if they expect to benefit from it, but again everything depends on the personality of the user and the application specifics. Thus we did not propose any numerical measures for predicting user acceptance or for evaluating trade-offs between the cost of adaptation and gain in performance. Instead, we presented qualitative guidelines for lightweight adaptation design and suggested relying on feedback from each user for deciding whether lightweight adaptation is acceptable or not.

Although each of the proposed adaptation approaches was tested in one application domain only, researchers in other domains may benefit from these studies, too. The approaches to adaption to social context and to the transfer of knowledge from one context to another, for example, may be applicable to a broader range of systems, e.g. it has been suggested in a survey of concept drift adaptation [Gama 2013] and in a survey of situation identification techniques [Ye 2012] that knowledge transfer can be a potential line of research. As the design guidelines proposed for lightweight adaptation were developed after an analysis of the state of the art and our own work in several application domains, they may be suitable for various personal applications. Thus our work may facilitate the making of more intelligent personal applications.

References

[Aarts 2009] Aarts, E., de Ruyter, B., New research perspectives on Ambient Intelligence, Journal of Ambient Intelligence and Smart Environments, 2009, Volume 1, Number 1, pp. 5-14.

[Adomavicius 2005] Adomavicius, G., Sankaranarayanan, R., Sen, S., Tuzhilin, A., Incorporating contextual information in recommender systems using a multidimensional approach, ACM Transactions on Information Systems, 23(1):103–145, 2005.

[Adomavicius 2007] Adomavicius, G., Kwon, Y., New recommendation techniques for multicriteria rating systems, IEEE Intell. Syst. 22(3), 48–55 (2007).

[Adomavicius 2011] Adomavicius, G., Tuzhilin, A., Context-Aware Recommender Systems, in Recommender Systems Handbook, 2011, pp. 217-253.

[Ahn 2006] Ahn, H., Kim, K.-j., Han, I., Mobile Advertisement Recommender System Using Collaborative Filtering: MAR-CF, Technical Report of the Korea Society of Management Information Systems (2006), pp. 709-715.

[Allano 2010] Allano, L., Dorizzi, B., Garcia-Salicetti, S., Tuning cost and performance in multibiometric systems: A novel and consistent view of fusion strategies based on the Sequential Probability Ratio Test (SPRT), Pattern Recognition Letters 31 (2010), 884–890.

[Anand 2010] Anand, D., Bharadwaj, K., Adaptive User Similarity Measures for Recommender Systems: A Genetic Programming Approach, in Proceedings of ICCSIT (2010), 121-125.

[Apeh 2013] Apeh, E., Gabrys, B., Detecting and Visualizing the Change in Classification of Customer Profiles based on Transactional Data, Evolving Systems, March 2013, Volume 4, Issue 1, pp. 27-42.

[Aste 2015] Aste, M., Boninsegna, M., Freno, A., Trentin, E., Techniques for dealing with incomplete data: a tutorial and survey, Pattern Analysis and Applications, February 2015, Volume 18, Issue 1, pp. 1-29.

[Atrey 2010] Atrey, P. K., Hossain, M. A., El Saddik, A., Kankanhalli, M. S., Multimodal Fusion for Multimedia Analysis: a Survey, Multimedia Systems 16, pp. 345-379, 2010.

[Baltrunas 2012] Baltrunas, L., Ludwig, B., Peer, S., Ricci, F., Context relevance assessment and exploitation in mobile recommender systems, Personal Ubiquitous Computing 16, 5 (2012), pp. 507-526.

[Baltrunas 2014] Baltrunas, L., Ricci, F., Experimental evaluation of context-dependent collaborative filtering using item splitting, User Modeling and User-Adapted Interaction 24, 1-2 (February 2014), 7-34.

[Bengio 2002] Bengio, S., Marcel, C., Marcel, S., Mariethoz, J., Confidence Measures for Multimodal Identity Verification, Information Fusion, Vol. 3, No. 4, pp. 267-276, 2002.

[Bengio 2009] Bengio, Y., Learning Deep Architectures for AI, Foundations and Trends in Machine Learning 2, 1 (2009), pp. 1-127.

[Berkovsky 2008] Berkovsky, Sh., Kuflik, T., Ricci, F., Mediation of User Models for Enhanced Personalisation in Recommender Systems, UMUAI (2008) 18: 245-286.

[Bharadwaj 2014] Bharadwaj, S., Vatsa, M., Singh, R., Aiding face recognition with social context association rule based re-ranking, in Biometrics (IJCB), 2014 IEEE International Joint Conference on, pp.1-8, Sept. 29 2014-Oct. 2 2014.

[Bhatt 2011] Bhatt, Ch., Kankanhalli, M., Multimedia data mining: state of the art and challenges, Multimedia Tools Appl., 51(1), pp. 35-76 (2011).

[Blanco-Fernandez 2010] Blanco-Fernandez, Y., Lopez-Nores, M., Pazos-Arias, J.J., Gil-Solla, A., Ramos-Cabrer, M., Exploiting digital TV users' preferences in a tourism recommender system based on semantic reasoning, IEEE Trans. on Consumer Electronics, 56 (2), 904-912 (2010).

[Bohmer 2010] Böhmer, M., Bauer, G., Exploiting the Icon Arrangement on Mobile Devices as Information Source for Context-awareness, Mobile HCI (2010).

[Borras 2014] Borràs, J., Moreno, A., Valls, A., Intelligent tourism recommender systems: A survey, Expert Systems with Applications 11/2014; 41(16):7370–7389.

[Brezeale 2008] Brezeale, D., Cook, D.J., Automatic Video Classification: A Survey of the Literature, IEEE Trans. on Systems, Man, and Cybernetics, Part C: Applications and Reviews, 38 (3), pp. 416-430, 2008.

[Britto 2014] Britto, A.S., Jr., Sabourin, R., de Oliveira, L.E.S., Dynamic selection of classifiers – a comprehensive review, Pattern Recognition 47 (11) (2014) 3665–3680.

[Calumby 2012] Calumby, R., Torres, R., Gonçalves, M., Multimodal retrieval with relevance feedback based on genetic programming, Multimedia Tools and Applications, 2012, pp. 1-29.

[Calvo 2010] Calvo, R., D'Mello, S., Affect Detection: An Interdisciplinary Review of Models, Methods, and Their Applications, IEEE Transactions on Affective Computing 1 (1), pp. 18-37, 2010.

[Canuto 2013] Canuto, A.M.P., Pintro, F., Xavier Junior, J.C., Investigating fusion approaches in multi-biometric cancellable recognition, Expert Syst. Appl. 40(6):1971-1980 (2013).

[Cao 2010] Cao, L., Domain-driven data mining: challenges and prospects, IEEE Transactions on Knowledge and Data Engineering, Vol. 22 (6), pp. 755–769, (2010).

[Capuano 2015] Capuano, N., D'Aniello, G., Gaeta, A., Miranda, S., A personality based adaptive approach for information systems, Computers in Human Behavior 44: 156-165 (2015).

[Caridakis 2008] Caridakis, G., Karpouzis, K., Kollias, S., User and context adaptive neural networks for emotion recognition, Neurocomputing, 71 (13-15), pp. 2553-2562, 2008.

[Casale 2012] Casale, P., Pujol, O., Radeva, P., Personalization and user verification in wearable systems using biometric walking patterns, Personal Ubiquitous Computing 16, 5 (June 2012), 563-580.

[Casale 2015] Casale, P., Altini, M., Amft, O., Transfer Learning in Body Sensor Networks using Ensembles of Randomised Trees, IEEE Internet of Things Journal, 2015, 2(1), pp. 33-40.

[Cavalin 2012] Cavalin, P., Sabourin, R., Suen, Ch., LoGID: An adaptive framework combining local and global incremental learning for dynamic selection of ensembles of HMMs, Pattern Recognition 45(9): 3544-3556 (2012).

[Chen 2008] Chen, Y.-L., Cheng, L.-C., Chuang, C.-N., A group recommendation system with consideration of interactions among group members, Expert Systems with Applications, 34(3):2082-2090, 2008.

[Chen 2013] Chen, J., Liu, X., Tu, P., ragones, A., Learning person-specific models for facial expression and action unit recognition, Pattern Recognition Letters 34 (2013) 1964–1970.

[Cheng 2009] Cheng, E., Jing, F., Zhang, L., A unified relevance feedback framework for web image retrieval, IEEE Transactions on Image Processing 2009,18 (6), pp. 1350-1357.

[Christou 2012] Christou, I.T., Gekas, G., Kyrikou, A., A classifier ensemble approach to the TV-viewer profile adaptation problem, International Journal of Machine Learning and Cybernetics (2012), Vol. 3, pp. 313-326.

[Connolly 2013] Connolly, J.F., Granger, E., Sabourin, R., Dynamic multi-objective evolution of classifier ensembles for video face recognition, Applied Soft Computing Volume 13, Issue 6, June 2013, pp. 3149-3166.

[Da Silva 2012] Da Silva, S. F., Alves, L.G.P., Bressan, G., Personal TVware: An Infrastructure to Support the Context-Aware Recommendation for Personalized Digital TV, International Journal of Computer Theory and Engineering, Vol. 4, No. 2, 2012, pp. 131-136.

[De Moor 2011] De Moor, K., De Pessemier, T., Mechant, P., Courtois, C., De Marez, A., Martens, L., Users' (Dis)satisfaction with the personalTV application: Combining objective and subjective data, ACM Computers in Entertainment 9(3): 18 (2011).

[Deravi 2012] Deravi, F., Intelligent Biometrics, in Second Generation Biometrics: The Ethical, Legal and Social Context, the International Library of Ethics, Law and Technology Volume 11, 2012, pp. 177-191.

[Douglas-Cowie 2007] Douglas-Cowie, E. et al., The HUMAINE Database: Addressing the Collection and Annotation of Naturalistic and Induced Emotional Data. Proc. ACII 2007, LNCS 4738: pp. 488-500, 2007.

[Dourish 2004] Dourish, P., What we talk about when we talk about context. Personal and ubiquitous computing, 8(1):19–30, 2004.

[Dumas 2009] Dumas, B., Lalanne, D., Oviatt, Sh., Multimodal Interfaces: A Survey of Principles, Models and Frameworks, In Human Machine Interaction, Eds. Lalanne, D. and Kohlas, J., Springer 2009, pp. 3-26.

[Dumas 2013] Dumas, B., Solórzano, M., Signer, B., Design guidelines for adaptive multimodal mobile input solutions, MobileHCI 2013, ACM, pp. 285-294

[Erzin 2005] Erzin, E., Yemez, Y., Tekalp, A. M., Multimodal Speaker Identification Using an Adaptive Classifier Cascade Based on Modality Reliability, IEEE Transactions on Multimedia, Vol. 7, No. 5, pp. 840-852, October 2005.

[Evers 2014] Evers, Ch., Kniewel, R., Geihs, K., Schmidt, L., The user in the loop: Enabling user participation for self-adaptive applications, Future Generation Computer Systems, Volume 34, May 2014, pp. 110-123.

[Evgeniou 2004] Evgeniou, T., Pontil, M., Regularized multi-task learning, In Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining (KDD 2004), pp. 109-117.

[Fatukasi 2007] Fatukasi, O., Kittler, J., Poh, N., Quality Controlled Multimodal Fusion of Biometric Experts, CIARP 2007, pp. 881–890.

[Fatukasi 2008] Fatukasi, O., Kittler, J., Poh, N., Estimation of Missing Values in Multimodal Biometric Fusion, IEEE Conf. on Biometrics, 2008, pp. 1-6.

[Ferecatu 2008] Ferecatu, M., Boujemaa, N., Crucianu, M., Semantic interactive image retrieval combining visual and conceptual content description, ACM Multimedia Systems Journal, Vol. 13, No. 5-6, p. 309-322, 2008.

[Fierrez-Aguilar 2004] Fierrez-Aguilar, J., Ortega-Garcia, J., Gonzalez-Rodriguez, J., Bigun, J., Kernel-Based Multimodal Biometric Verification Using Quality Signals, in Defense and Security Symposium, Proc. of SPIE, 5404, pp. 544–554, 2004.

[Findlater 2008] Findlater, L., Mcgrenere, J., Impact of screen size on performance, awareness, and user satisfaction with adaptive graphical user interfaces, In Proc. of CHI 08, pages 1247-1256 (2008).

[Forbes-Riley 2004] Forbes-Riley, K., Litman, D. Predicting Emotion in Spoken Dialogue from Multiple Knowledge Sources. Proc. HLT/NAACL 2004, pp. 201-208, 2004.

[Gaber 2006] Gaber, M., Yu, P., A framework for resource-aware knowledge discovery in data streams: a holistic approach with its application to clustering. In: The 2006 ACM Symposium on Applied Computing (SAC).

[Gajos 2008] Gajos, K., Wobbrock, J., Weld, D., Improving the performance of motor-impaired users with automatically-generated, ability-based interfaces, in: CHI ’08: Proceedings of the Twenty-Sixth Annual SIGCHI Conference on Human Factors in Computing Systems, ACM, 2008, pp. 1257–1266.

[Gajos 2010] Gajos, K., Weld, D., Wobbrock, J., Automatically generating personalized user interfaces with Supple, Artificial Intelligence, 174:910-950, 2010.

[Gallagher 2008] Gallagher, A., Chen, T., Using context to recognize people in consumer images, IPSJ Journal 49, 1234–1245 (2008).

[Gama 2012] Gama J., A survey on learning from data streams: current and future trends, Progress in Artificial Intelligence, Volume 1, Issue 1, 2012, pp. 45-55.

[Gama 2013] Gama, J., Zliobaite, I., Bifet, A., Pechenizkiy, M., Bouchachia, A., A Survey on Concept Drift Adaptation, ACM Computing Surveys, Vol. 1, Article 1, 2013.

[Garcia 2014] García, I., Sebastia, L., A negotiation framework for heterogeneous group recommendation, Expert Systems with Applications 2014; 41(4):1245-1261.

[Garcia-Borroto 2014] García-Borroto, M., Martínez-Trinidad, J.F., Carrasco-Ochoa, J.A., A survey of emerging patterns for supervised classification, Artificial Intelligence Review, December 2014, Volume 42, Issue 4, pp 705-721.

[Ghorab 2013] Ghorab, M.R., Zhou, D., O'Connor, A., Wade, V., Personalised Information Retrieval: survey and classification, User Modelling User-Adapted Interaction (2013) 23:381-443.

[Gil 2012] Gil, M., Giner, P., Pelechano, V., Personalization for unobtrusive service interaction, Personal and Ubiquitous Computing, Vol. 16, Issue 5, 2012, pp. 543-561.

[Giot 2012] Giot, R., Rosenberger, Ch., Genetic programming for multibiometrics. Expert Syst. Appl. 39(2): 1837-1847 (2012).

[Griol 2014] Griol, D., Molina, J.M., Callejas, Z., Modeling the user state for context-aware spoken interaction in ambient assisted living, Applied Intelligence, June 2014, Volume 40, Issue 4, pp 749-771.

[Gunes 2010] Gunes, H., Pantic, M., Automatic, Dimensional and Continuous Emotion Recognition, International Journal of Synthetic Emotions, 1(1), pp. 68-99, 2010.

[Guz 2010] Guz, U., Tur, G., Hakkani-Tür, D., Cuendet, S., Cascaded model adaptation for dialog act segmentation and tagging, Computer Speech & Language, Volume 24, Issue 2, 2010, pp. 289-306.

[Haghighi 2009] Haghighi, P., Zaslavsky, A., Krishnaswamy, S., Gaber, M., Loke, S., Context-aware adaptive data stream mining, Intelligent data analysis, 2009, Volume 13, Issue 3, pp. 423-434.

[Hariri 2012] Hariri, N., Mobasher, B., Burke, R., Context-aware music recommendation based on latent topic sequential patterns, In Proceedings of the ACM conference on Recommender Systems 2012, pp. 131-138.

[Holbling 2010] Hölbling, G., Pleschgatternig, M., Kosch, H., PersonalTV – A TV recommendation system using program metadata for content filtering, Multimedia Tools Appl. 46(2-3): 259-288 (2010).

[Hossain 2013] Hossain, M.A., Shirehjini, A.A., Alghamdi, A.S., Saddik, A., Adaptive interaction support in ambient-aware environments based on quality of context information, Multimedia Tools and Applications 67, 2 (November 2013), 409-432.

[Hu 2011] Hu, W., Xie, N., Li, L., Zeng, X., Maybank, S., A Survey on Visual Content-Based Video Indexing and Retrieval, IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, vol.41, no.6, pp.797-819, 2011.

[Hussein 2014] Hussein, T., Linder, T., Gaulke, W., Ziegler, J., Hybreed: A software framework for developing context-aware hybrid recommender systems, User Modeling and User-Adapted Interaction 24, 1-2 (February 2014), 121-174.

[Jain 2012] Jain, A., Kumar, A., Biometric Recognition: An Overview, in Second Generation Biometrics: The Ethical, Legal and Social Context, the International Library of Ethics, Law and Technology Volume 11, 2012, pp. 49-79.

[Jameson 2007] Jameson, A., Smyth, B.: Recommendation to Groups, In: Brusilovsky, P., Kobsa, A., Nejdl, W., (eds.): The Adaptive Web (2007), pp. 596-627.

[Jiang 2008] Jiang, W., Zavesky, E., Chang, Sh.-F., Loui, A., Cross-domain learning methods for high-level visual concept classification, IEEE International Conference on Image Processing, 2008, pp. 161-164.

[Jiang 2012] Jiang, Y.G., Dai, Q., Wang, J., Ngo, C.W., Xue, X., Chang, S.F., Fast semantic diffusion for large-scale context-based image and video annotation, IEEE Transactions on Image Processing, Vol. 21, No. 6 (2012).

[Joho 2011] Joho, H., Staiano, J., Sebe, N. and Joemon, M.J., Looking at the viewer: analysing facial activity to detect personal highlights of multimedia contents, Multimedia Tools and Applications, 51(2), 2011, pp. 505-523.

[Joshi 2012] Joshi, A., Porikli, F., Papanikolopoulos, N., Scalable active learning for multi-class image classification, IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), 2012, Vol. 34, No. 11, pp. 2259-2273.

[Katuka 2014] Katuka, J., Mohamad, D., Saba, T., El-Affendi, M., Mohamed, A.S., An Analysis of Object Appearance Information and Context Based Classification, 3D Research (2014), 5:24.

[Khaleghi 2013] Khaleghi, B., Khamis, A., Karray, F.O., Razavi, S.N., Multisensor data fusion: A review of the state-of-the-art, Information Fusion, Volume 14, Issue 1, January 2013, pp. 28-44.

[Khoury 2014] Khoury, E., El Shafey, L., McCool, Ch., Günther, M., Marcel, S., Bi-modal biometric authentication on mobile phones in challenging conditions, Image and Vision Computing, Volume 32, Issue 12, December 2014.

[Kirstein 2012] Kirstein, S., Wersing, H., Gross, H.-M., Körner, E., A life-long learning vector quantization approach for interactive learning of multiple categories, Neural Networks 28, pp. 90-105 (2012).

[Kittler 2007] Kittler J., Poh N., Fatukasi, O., Messer, K., Kryszczuk, K., Richiardi, J., Drygajlo, A., Quality Dependent Fusion of Intramodal and Multimodal Biometric Experts, Proc. SPIE 6539, Biometric Technology for Human Identification IV, 2007.

[Kong 2011] Kong, J., Zhang, W. Y., Yu, N., Xia, X. J., Design of human-centric adaptive multi-modal interfaces, Int. J. Hum.-Comput. Stud. 69, (2011), pp. 854-869.

[Kotti 2012] Kotti, M., Paternò, F., Speaker-independent emotion recognition exploiting a psychologically-inspired binary cascade classification schema, Int. J. Speech Technol. 15, 2 (June 2012), pp. 131-150.

[Krupitzer 2015] Krupitzer, C., Roth, F.M., VanSyckel, S., Schiele, G., Becker, C., A survey on engineering approaches for self-adaptive systems, Pervasive Mobile Computing, 17 (2015), pp. 184-206.

[Kumar 2010] Kumar, A., Kanhangad, V., Zhang, D., A new framework for adaptive multimodal biometrics management, IEEE Transactions on Information Forensics and Security, Volume 5 Issue 1, March 2010, pp. 92-102.

[Kumar 2013] Kumar, A., Hanmandlu, M., Gupta, H. M., Ant colony optimization based fuzzy binary decision tree for bimodal hand knuckle verification system, Expert Systems with Applications, Volume 40 Issue 2, February, 2013, pp. 439-449.

[Kuncheva 2004] Kuncheva L., Combining Pattern Classifiers, Methods and Algorithms, Wiley, 2004.

[Lemke 2015] Lemke, Ch., Budka, M., Gabrys, B., Metalearning: a survey of trends and technologies, Artificial Intelligence Review 44, 1 (June 2015), 117-130

[Li 2005] Li, X., Ji, Q., Active affective State detection and user assistance with dynamic Bayesian networks, IEEE Trans. on Systems, Man, and Cybernetics, Part A: Systems and Humans, 35 (1), pp. 93-105, 2005.

[Li 2012] Li, H., Shi, Y., Liu, Y., Hauptmann, A., Xiong, Z., Cross-domain video concept detection: A joint discriminative and generative active learning approach, Expert Systems with Applications 39(15): 12220-12228 (2012).

[Lopez-Cozar 2011] López-Cózar, R., Silovsky, J., Kroul, M., Enhancement of emotion detection in spoken dialogue systems by combining several information sources, Speech Communication, Volume 53, Issues 9–10, 2011, pp. 1210-1228.

[Lu 2009] Lu, L., Content Discovery from Composite Audio: An unsupervised approach, PhD Thesis. Delft Univ. of Technology, 2009, http://homepage.tudelft.nl/c7c8y/Theses/PhDThesisLieLu.pdf

[Luan 2011] Luan, H., Zheng, Y.-T., Wang, M., Chua, T.-S., VisionGo: Towards video retrieval with joint exploration of human and computer, Information Sciences 181, 19 (2011), pp. 4197-4213.

[Macias-Escriva 2013] Macías-Escrivá, F.D., Haber, R., del Toro, R., Hernandez, V., Self-adaptive systems: A survey of current approaches, research challenges and applications, Expert Systems with Applications, Volume 40, Issue 18, 15 December 2013, pp. 7267-7279.

[Macik 2014] Macik, M., Cerny, T., Slavik, P., Context-sensitive, cross-platform user interface generation, Journal on Multimodal User Interfaces, June 2014, Volume 8, Issue 2, pp 217-229.

[Masthoff 2006] Masthoff, J., Gatt, A., In pursuit of satisfaction and the prevention of embarrassment: affective state in group recommender systems, User Modeling and User-Adapted Interaction 16(3–4), pp. 281–319 (2006).

[Metallinou 2012] Metallinou, A., Wollmer, M., Katsamanis, A., Eyben, F., Schuller, B., Narayanan, S, Context-Sensitive Learning for Enhanced Audiovisual Emotion Classification, IEEE Trans. on Affective Computing, 3(2), 2012, 184-198.

[Mondal 2015] Mondal, S., Bours, P., A computational approach to the continuous authentication biometric system, Information Sciences 304 (2015) 28-53.

[Morvant 2012] Morvant, E., Habrard, A., Ayache, S., Parsimonious unsupervised and semi-supervised domain adaptation with good similarity functions, Knowledge and Information Systems, 33(2), 2012, pp. 309-349.

[Mporas 2011] Mporas, I., Ganchev, T., Kocsis, O., Fakotakis, N., Context-adaptive pre-processing scheme for robust speech recognition in fast-varying noise environment, Signal Processing, 91 (8), 2011, pp. 2101-2111.

[Mukherjee 2014] Mukherjee, S., Pal, K., Majumder, B.P., Saha, C., Panigrahi, B.K., Das, S., Differential evolution based score level fusion for multi-modal biometric systems, Computational Intelligence in Biometrics and Identity Management (CIBIM), 2014 IEEE Symposium on, pp.38-44, 9-12 Dec. 2014.

[Nageshkumar 2009] Nageshkumar, M., Mahesh, P. K., Shanmukha Swamy, M. N., An Efficient Secure Multimodal Biometric Fusion Using Palmprint and Face Image, IJCSI International Journal of Computer Science Issues, Vol. 2, 2009, pp. 49-53.

[Nandakumar 2009] Nandakumar, K., Jain, A., Ross, A., Fusion in Multibiometric Identification Systems: What about the Missing Data? in Proceedings of 3rd IAPR/IEEE International Conference on Biometrics, pp. 743-752, June 2009, Alghero, Italy.

[Nguyen 2012] Nguyen, L., Odobez, J.-M., Gatica-Perez, D., Using self-context for multimodal detection of head nods in face-to-face interactions, ACM international conference on Multimodal Interaction 2012, pp. 289-292.

[Octavia 2011] Octavia, J.R., Raymaekers, Ch., Coninx, K., Adaptation in virtual environments: conceptual framework and user models, Multimedia Tools and Applications 54, 1 (August 2011), 121-142.

[Oku 2006] Oku, K., Nakajima, S., Miyazaki, J., Uemura, S., Context-Aware SVM for Context-Dependent Information Recommendation, in Proceedings of International Conference on Mobile Data Management 2006, pp. 109-112.

[Otsuka 2009] Otsuka, I., Shipman, S., Divakaran, A., A Video Browsing Enabled Personal Video Recorder, Multimedia Content Analysis, pp. 1-12, 2009.

[Palmisano 2008] Palmisano, C., Tuzhilin, A., Gorgoglione, M., Using Context to Improve Predictive Modeling of Customers in Personalization Applications, IEEE Transactions on Knowledge and Data Engineering, vol.20, no.11, pp.1535-1549, 2008.

[Pan 2010] Pan, S., Yang, Q., A Survey on Transfer Learning, IEEE Transactions on Knowledge and Data Engineering, vol. 22, no. 10, pp. 1345-1359, 2010.

[Panniello 2009] Panniello, U., Tuzhilin, A., Gorgoglione, M., Palmisano, C., Pedone, A., Experimental comparison of pre- vs. post-filtering approaches in context-aware recommender systems, in Proceedings of the ACM conference on Recommender systems, 2009, pp. 265-268.

[Panniello 2014] Panniello, U., Tuzhilin, A., Gorgoglione, M., Comparing context-aware recommender systems in terms of accuracy and diversity, User Modeling and User-Adapted Interaction 24, 1-2 (February 2014), 35-65.

[Papadopoulos 2011] Papadopoulos, G., Mezaris, V., Kompatsiaris, I., Strintzis, M. G., Joint modality fusion and temporal context exploitation for semantic video analysis. EURASIP Journal on Advances in Signal Processing, 2011, 89.

[Patel 2015] Patel, V. M., Gopalan, R., Li, R., Chellappa, R., Visual domain adaptation: a survey of recent advances, IEEE Signal Processing Magazine, May 2015.

[Patricia 2014] Patricia, N., Caputo, B., Learning to Learn, from Transfer Learning to Domain Adaptation: A Unifying Perspective. CVPR 2014: 1442-1449.

[Pauplin 2010] Pauplin, O., Caleb-Solly, P., Smith, J., User-centric image segmentation using an interactive parameter adaptation tool, Pattern Recognition 43, 2 (February 2010), 519-529.

[Poh 2007] Poh, N., Heusch, G., Kittler, J., On Combination of Face Authentication Experts by a Mixture of Quality Dependent Fusion Classifiers, in LNCS 4472, Multiple Classifiers System (MCS), Prague, 2007, pp. 344-356.

[Poh 2009] Poh, N., Wong, R., Kittler, J., Roli, F., Challenges and Research Directions for Adaptive Biometric Recognition Systems, IEEE Conf. on Biometrics 2009, pp. 753-764.

[Poh 2012] Poh, N., Kittler, J., A Unified Framework for Biometric Expert Fusion Incorporating Quality Measures, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 34 (1), 2012, pp. 3-18.

[Qi 2008] Qi, G.-J., Tang, J., Wang, M., Hua, X.-S., Rui, Y., Mei, T., Zhang, H.-J., Correlative multilabel video annotation with temporal kernels, ACM Trans. Multimedia Comput. Commun. Appl. 5, 1, Article 3 (2008), 27 pages.

[Radhakrishnan 2006] Radhakrishnan, R., Divakaran, A., Xiong, Z., Otsuka, I., A Content-Adaptive Analysis and Representation Framework for Audio Event Discovery from “Unscripted” Multimedia, EURASIP Journal on Applied Signal Processing, Vol. 2006, pp.1-24, 2006.

[Rafeh 2012] Rafeh, R., Bahrehmand, A., An adaptive approach to dealing with unstable behaviour of users in collaborative filtering systems, Journal of Information Science 38(3): 205-221 (2012).

[Rehman 2014] Rehman, A., Saba, T., Features extraction for soccer video semantic analysis: current achievements and remaining issues, Artificial Intelligence Review 41(3): 451-461, 2014.

[Reis 2008] Reis, T., de Sá, M., Carriço, L., Multimodal Interaction: Real Context Studies on Mobile Digital Artefacts, in Proceedings of Haptic and Audio Interaction Design HAID 2008.

[Roli 2008] Roli, F., Didaci, L., Marcialis, G. L., Adaptive biometric systems that can improve with use, Advances in Biometrics: Sensors, Systems and Algorithms, Springer, pp. 447-471 (2008).

[Ronkainen 2010] Ronkainen, S., Koskinen, E., Liu, Y., Korhonen, P., Environment Analysis as a Basis for Designing Multimodal and Multidevice User Interfaces, Human–Computer Interaction, 2010, Vol. 25, No 2, pp. 148-193.

[SAL] http://emotion-research.net/toolbox/toolboxdatabase.2006-09-26.5667892524

[Salamo 2012] Salamó, M., McCarthy, K., Smyth, B., Generating recommendations for consensus negotiation in group personalization services, Personal and Ubiquitous Computing June 2012, Volume 16, Issue 5, pp. 597-610.

[Schuller 2009] Schuller, B., Müller, R., Eyben, F., Gast, J., Hörnler, B., Wöllmer, M., Rigoll, G., Höthker, A., Konosu, H., Being bored? Recognising natural interest by extensive audiovisual integration for real-life application, Image and Vision Computing, 27, pp. 1760-1774, 2009.

[Schuller 2010] Schuller, B., Vlasenko, B., Eyben, F., Wollmer, M., Stuhlsatz, A., Wendemuth, A., Rigoll, G., Cross-Corpus Acoustic Emotion Recognition: Variances and Strategies, IEEE Trans. Affective Computing, 1 (2), pp. 119-131, 2010.

[Schwenker 2014] Schwenker, F., Trentin, E., Pattern classification and clustering: a review of partially supervised learning approaches, Pattern Recognition Letters 37:4-14, 2014.

[Senot 2010] Senot, Ch., Kostadinov, D., Bouzid, M., Picault, J., Aghasaryan, A., Bernier, C., Analysis of Strategies for Building Group Profiles, Proceedings of UMAP 2010.

[Settles 2009] Settles, B., Active Learning Literature Survey, Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009.

[Shih 2009] Shih, H.-Ch., Hwang, J.-N., Huang, Ch.-L., Content-Based Attention Ranking Using Visual and Contextual Attention Model for Baseball Videos, IEEE Transactions on Multimedia, Vol. 11, No. 2, pp. 244-255, 2009.

[Shin 2009] Shin, Ch., Woo, W., Socially Aware TV Program Recommender for Multiple Viewers, IEEE Transactions on Consumer Electronics 55(2), pp. 927-932 (2009).

[Shivappa 2010] Shivappa, S.T., Trivedi, M.M., Rao, B.D., Audiovisual Information Fusion in Human–Computer Interfaces and Intelligent Environments: A Survey, Proceedings of the IEEE, Vol. 98, No.10, pp. 1692-1715, 2010.

[Shyu 2008] Shyu, M.-L., Xie, Z., Chen, M., Chen, Sh.-Ch., Video Semantic Event/Concept Detection Using a Subspace-Based Multimedia Data Mining Framework, IEEE Transactions on Multimedia, 10 (2), pp. 252-259, 2008.

[Sim 2007] Sim, T., Zhang, Sh., Janakiraman, R., Kumar, S., Continuous Verification Using Multimodal Biometrics, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 29, Issue 4, April 2007, pp. 687-700.

[Sim 2014] Sim, H. M., Asmuni, H., Hassan, R., Othman, R. M., Multimodal biometrics: Weighted score level fusion based on non-ideal iris and face images, Expert Systems with Applications 41 (2014) 5390-5404.

[Snidaro 2015] Snidaro, L., García, J., Llinas, J., Context-based Information Fusion: A survey and discussion, Information Fusion, Vol. 25, September 2015, pp. 16-31.

[Sotelo 2009] Sotelo, R., Blanco-Fernández, Y., López-Nores, M., Gil-Solla, A., Pazos Arias, J., TV program recommendation for groups based on muldimensional TV-anytime classifications, IEEE Transactions on Consumer Electronics 55(1): 248-256 (2009).

[Stober 2013] Stober, S., Nürnberger, A., Adaptive music retrieval – a state of the art, Multimedia Tools and Applications, Vol. 65, Issue 3, 2013, pp. 467-494.

[Syed 2014] Syed, Z., Banerjee, S., Cukic, B., Continual authentication, Biometric Technology Today, Vol. 2014, Issue 6, June 2014, pp. 5-9.

[Takahashi 2004] Takahashi, K., Mimira, M., Isobe, Y., Seto, Y., A Secure and User-Friendly Multimodal Biometric System, in Biometric Technology for Human Identification, edited by A. Jain, N. Ratha, Proceedings of SPIE Vol. 5404, August 2004.

[Tang 2012] Tang, Sh., Zheng, Y.-T., Wang, Y., Chua, T.-S., Sparse Ensemble Learning for Concept Detection, IEEE Transactions on Multimedia, Vol. 14, No. 1, pp. 43-54, 2012.

[Tawari 2010] Tawari, A., Trivedi, M., Speech Emotion Analysis: Exploring the Role of Context, IEEE Transactions on Multimedia 12(6), 2010.

[Thomee 2012] Thomee, B., Lew, M., Interactive search in image retrieval: a survey, International Journal of Multimedia Information Retrieval, Vol. 1(2), 2012, pp. 71-86.

[Thyagaraju 2011] Thyagaraju, G.S., Kulkarni, U.P., Family Aware TV Program and Settings Recommender, International Journal of Computer Applications, 29(4), 2011, pp. 1-18.

[Turk 2014] Turk, M., Multimodal interaction: A review, Pattern Recognition Letters 36 (January 2014), 189-195.

[Vera-Rodriguez 2012] Vera-Rodriguez, R., Tome, P., Fierrez, J., Ortega-Garcia, J., Fusion of footsteps and face biometrics on an unsupervised and uncontrolled environment, Proceedings of the SPIE, Volume 8371, 2012.

[Vinciarelli 2012] Vinciarelli, A., Pantic, M., Heylen, D., Pelachaud, C., Poggi, I., D'Errico, F., Schroeder, M., Bridging the Gap between Social Animal and Unsocial Machine: A Survey of Social Signal Processing, IEEE Trans. on Affective Computing, Vol. 3, Issue 1, pp. 69-87.

[Wagner 2011] Wagner, J., Lingenfelser, F., Andre, E., Kim, J., Vogt, T., Exploring Fusion Methods for Multimodal Emotion Recognition with Missing Data, IEEE Transactions on Affective Computing, Vol. 2, No. 4, pp. 206-218, 2011.

[Wang J 2011] Wang, J., Li, Y., Wang, C., How to handle missing data in robust multi-biometrics verification, Int. J. Biometrics, Vol. 3, No. 3, 2011, pp. 265-283.

[Weng 2008] Weng, M.-F., Chuang, Y.-Y., Multi-cue fusion for semantic video indexing, in Proc. of the ACM Multimedia, 2008, pp. 71-80.

[Weng 2012] Weng, M.-F., Chuang, Y.-Y., Cross-Domain Multicue Fusion for Concept-Based Video Indexing, IEEE Trans. Pattern Anal. Mach. Intell. 34, 10 (2012), pp. 1927-1941.

[Wollmer 2010] Wollmer, M., Metallinou, A., Eyben, F., Schuller, B., Narayanan, S., Context-Sensitive Multimodal Emotion Recognition from Speech and Facial Expression using Bidirectional LSTM Modeling, Proc. Interspeech 2010, pp. 2362-2365, 2010.

[Wright 2008] Safeguards in a World of Ambient Intelligence, Eds. Wright, D., Gutwirth, S., Friedewald, M., Vildjiounaite, E., Punie, Y., Springer, Jan. 2008.

[Wu 2007] Wu, J., Huo, Q., A Study of Minimum Classification Error (MCE) Linear Regression for Supervised Adaptation of MCE-Trained Continuous-Density Hidden Markov Models, IEEE Trans. on Audio, Speech, and Language Processing, 15 (2), pp. 478-488, 2007.

[Xu 2006] Xu, H., Chua, T.-S., Fusion of AV Features and External Information Sources for Event Detection in Team Sports Video, ACM Transactions on Multimedia Computing, Communications and Applications, Vol. 2, No. 1, 2006, pp. 44-67.

[Xu 2008] Xu, M., Xu, C., Duan, L., Jin, J. S., Luo, S., Audio keywords generation for sports video analysis, ACM Trans. on Multimedia Computing, Communications, and Applications (TOMCCAP), 4 (2), pp. 1-23, 2008.

[Xu 2014] Xu, J.Y., Chang, H.I., Chien, C., Kaiser, W.J., Pottie, G.J., Context-driven, prescription-based personal activity classification: methodology, architecture, and end-to-end implementation, IEEE Journal of Biomedical and Health Informatics 2014 May;18(3):1015-25.

[Yang 2007] Yang, J., Yan, R., Hauptmann, A., Cross-domain video concept detection using adaptive SVMs, in Proceedings of the 15th international conference on Multimedia 2007, pp. 188-197.

[Yang 2009] Yang, J., A General Framework for Classifier Adaptation and its Applications in Multimedia, Ph.D thesis, 2009.

[Yang 2012] Yang, J., Tong, W., Hauptmann, A., A Framework for Classifier Adaptation for Large-Scale Multimedia Data, Proceedings of the IEEE, 100(9), 2012, pp. 2639-2657.

[Yao 2012] Yao, T., Ngo, C. W., Zhu, S. A., Predicting Domain Adaptivity: Redo or Recycle?, in Proceedings of ACM Multimedia 2012, pp. 821-823.

[Yasumura 2007] Yasumura, Y., Kitani, N., Uehara, K., Quick adaptation to changing concepts by sensitive detection, In Proceedings of the 20th international conference on Industrial, engineering, and other applications of applied intelligent systems (2007), pp. 855-864.

[Ye 2012] Ye, J., Dobson, S., McKeever, S., Situation identification techniques in pervasive computing: a review, Pervasive and Mobile Computing, 8 (2012), pp. 36-66.

[Yin 2005] Yin, P.-Y., Bhanu, B., Chang, K.-Ch., Dong, A., Integrating Relevance Feedback Techniques for Image Retrieval Using Reinforcement Learning, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005, Vol. 27, No. 10, pp. 1536-51.

[Yin 2010] Yin, P.-Y., Particle Swarm Optimization for Automatic Selection of Relevance Feedback Heuristics, ICSI (1) 2010: 167-174.

[Yu 2006] Yu, Z., Zhou, X., Hao, Y., Gu, J., TV program recommendation for multiple viewers based on user profile merging, UMUAI (2006) 16: 63-68.

[Yu 2009] Yu, Zh., Nam, M. Y., Rhee, P. K., Online evolutionary context-aware classifier ensemble framework for object recognition, IEEE International Conference on Systems, Man and Cybernetics, 2009, pp. 3428-3433.

[Zeng 2009] Zeng, Z., Pantic, M., Roisman, G., Huang, T., A Survey of Affect Recognition Methods: Audio, Visual, and Spontaneous Expressions. IEEE Trans. on PAMI, 31(1): pp. 39-58, 2009.

[Zhang 2005] Zhang, D., Gatica-Perez, D., Bengio, S., McCowan, I., Semi-supervised Meeting Event Recognition with Adapted HMMs, Proc. ICME 2005, pp. 611-618, 2005.

[Zhang 2009] Zhang, J., Ye, L., Content based image retrieval using unclean positive examples, IEEE Transactions on Image Processing 18 (10), October 2009, pp. 2370-2375.

[Zhang 2014] Zhang, L., Tan, T., Ross, A., Zafeiriou, S., Special issue on "Multi-biometrics and Mobile-biometrics: Recent Advances and Future Research", Image and Vision Computing, Vol. 32, Issue 12, December 2014, pp. 1145-1146.

[Zhou 2010] Zhou, Zh.-H., Li, M., Semi-supervised learning by disagreement, Knowledge and Information Systems, Vol. 24 (3) 2010, pp. 415-439.

[Zhu 2007] Zhu, G., Huang, Q., Xu, Ch., Xing, L., Gao, W., Yao, H., Human Behavior Analysis for Highlight Ranking in Broadcast Racket Sports Video, IEEE Trans. on Multimedia, 9 (6), pp. 1167-1182, 2007.

[Zliobaite 2012] Žliobaitė, I., Bakker, J., Pechenizkiy, M., Beating the baseline prediction in food sales: How intelligent an intelligent predictor is?, Expert Systems with Applications, Volume 39, Issue 1, 2012, pp. 806-815.

[Zliobaite 2015] Žliobaitė, I., Budka, M., Stahl, F., Towards cost-sensitive adaptation: when is it worth updating your predictive model?, Neurocomputing 150(A), p. 240-249.

PUBLICATION I

Empirical evaluation of combining unobtrusiveness and security requirements in multimodal biometric systems

Image and Vision Computing,
Vol. 27 (2009), No. 3, pp. 279–292.

Available online at www.sciencedirect.com

ScienceDirect

Image and Vision Computing 27 (2009) 279–292

www.elsevier.com/locate/imavis

Empirical evaluation of combining unobtrusiveness and security requirements in multimodal biometric systems

Elena Vildjiounaite *, Vesa Kyllönen, Heikki Ailisto

Technical Research Centre of Finland, Mobile Interaction, Kaitoväylä 1, P.O. Box 1100, 90571 Oulu, Finland

Received 16 January 2006; received in revised form 5 October 2007; accepted 16 December 2007

Abstract

Unobtrusive user authentication is more convenient than explicit interaction and can also increase system security because it can be performed frequently, unlike the current "once explicitly and for a long time" practice. Existing unobtrusive biometrics (e.g., face, voice, gait) do not perform sufficiently well for high-security applications, however, while reliable biometric authentication (e.g., fingerprint or iris) requires explicit user interaction. This work presents experiments with a cascaded multimodal biometric system, which first performs unobtrusive user authentication and requires explicit interaction only when the unobtrusive authentication fails. Experimental results obtained for a database of 150 users show that even with a fairly low performance of unobtrusive modalities (Equal Error Rate above 10%), the cascaded system is capable of satisfying a security requirement of a False Acceptance Rate less than 0.1% with an overall False Rejection Rate of less than 0.2%, while authenticating unobtrusively in 65% of cases.

Keywords: Biometrics; Multimodal fusion; Cascade

1. Introduction

Unobtrusive authentication would be convenient for users in many situations. Effortless computer login or opening of an office/home door, for example, would be more convenient than touching a fingerprint sensor or interacting with an iris recognition sensor. Furthermore, unobtrusive authentication can be performed more frequently than explicit authentication, which makes the systems more secure. If the authentication procedure requires explicit user interaction, it is usually performed only at login to a computer system, or during the "opening of a door" in other access control applications. Thus, the overall system security level decreases due to the existence of long time periods (many hours for desktop computers and office access control applications, and many weeks for mobile phones) when the user is assumed to be the same as during the authentication process, which might not be the case. This is well illustrated by the example of mobile phones: most of them are lost or stolen after login, when personal data are easily accessible. Since desktop computers and mobile devices nowadays contain a lot of personal information (such as images, contacts, plans stored in a user calendar, messages and emails) and the services accessed via them are becoming increasingly more sensitive (e.g., remote transactions, telework, email exchanges with friends, relatives, doctors, etc.), there is a growing need to protect computer systems continuously in a user-friendly way [1]. Similarly, office access control applications would be more secure if the identity of visitors were verified frequently and invisibly rather than through one explicit authentication at the entrance.

Unfortunately most reliable biometric methods (such as iris or fingerprint authentication) require explicit user interaction, while methods which can be used for

* Corresponding author. Tel.: +358 40 725 2470.
E-mail addresses: elena.vildjiounaite@vtt.fi (E. Vildjiounaite), vesa.kyllonen@vtt.fi (V. Kyllönen), heikki.ailisto@vtt.fi (H. Ailisto).
URL: http://www.vtt.fi.

0262-8856/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.imavis.2007.12.001

unobtrusive authentication (such as face, voice and gait recognition) do not have sufficiently high recognition rates to be used as the only means of authentication. Furthermore, the recognition rates of unobtrusive biometric modalities are highly dependent on environmental conditions: speaker recognition is vulnerable to noise, and video-based biometrics depend on lighting. The survey of Jain et al. [2] presents state-of-the-art error rates for face and voice modalities under various conditions as follows: the False Rejection Rate (FRR) for face recognition is about 10% and the False Acceptance Rate (FAR) is about 1%, while for voice recognition the FRR is about 5–10% and FAR is about 2–5%. Multimodal fusion usually improves recognition rates, but the performance of unobtrusive biometrics under uncontrolled conditions is nevertheless fairly poor. In experiments with the fusing of face and voice data, collected in uncontrolled adverse conditions as a part of the BANCA database, for example, the Equal Error Rate for multimodal fusion was above 4% [3].

This paper proposes a method for reducing user effort and at the same time keeping recognition rates within the desirable limits. This is achieved by combining unobtrusive user verification with a more reliable biometric modality in a cascaded system that first attempts to perform unobtrusive verification and requires explicit user effort only in cases of its failure. In an office access control application, it would be convenient for users if the office door opened when the user approached, requiring interaction with fingerprint or iris sensors only in rare cases, e.g., if the user's voice changes because of flu, or when darkness in the corridor hinders face recognition. For mobile devices and desktop computers, depending on the application requirements, a cascaded system could require explicit authentication either after implicit authentication has failed once or several times; or during certain time interval; or in a certain context, e.g., if a user opens an application which requires a higher security level (such as a banking application or the copying of sensitive data), or if a mobile device finds itself in a strange location.

To the best of our knowledge, research into multimodal biometric fusion rarely states the need to reduce user effort as an important goal. Researchers compare the performances of different fusion methods, e.g., trained and fixed rules [4], and suggest novel fusion schemes such as user-dependent fusion (learning separate models for each user) [5,6] or fusion utilising confidence measures of modalities [7]. Most of the work with fusion has been concerned with parallel system architectures (using all modalities simultaneously), however, and unobtrusive and obtrusive biometric modalities are often used together, e.g., in experiments with the simultaneous use of fingerprint and face modalities [5,8] or the simultaneous use of audio–visual and handwritten signature modalities [9].

Although cascaded systems have been used successfully in image processing tasks for improving classification performance, it is still uncommon to use them in multimodal biometric fusion. In the case of image processing an improvement in performance can be achieved by splitting a difficult problem into several easier subproblems and solving them one by one, by partitioning the data set. In the work of Huang [10] this is done by leaving negative training examples which were not classified correctly by the current stage for the next stage. All the stages in such cascaded systems use the same set of features, however, unlike multimodal biometric systems, which should use different sets of features (provided by different biometric modalities) at different stages.

In multimodal biometric fusion, cascaded systems were initially suggested for increasing the operating speeds of identification systems, as in the work of Hong et al. [11], where a multimodal identification system first finds several best matches for one modality and then searches for the best match for the second modality only among these existing matches. A. Jain [12] states in his "Introduction to Biometric Recognition" that in the serial (cascaded) mode of operation "the output of one biometric trait is typically used to narrow down the number of possible identities before the next trait is used", thus showing that cascaded fusion for verification purposes has not received much attention among researchers.

The situation has started to change only recently, with the suggestion by Takahashi et al. [13] of a cascaded multimodal system, that allows users to choose the order of the modalities, thus increasing the user-friendliness and population coverage (people who have problems with a certain biometric modality are free to choose another modality first), but can also facilitate spoofing. Takahashi et al. [13] propose a Sequential Probability Ratio Test for use in multimodal decision fusion, and prove the ability of the proposed method to keep FAR within the desired limits in experiments on a database of five people. Takahashi et al. do not, however, present any results indicative of overall system performance (the FRR which can be achieved with different system configurations).

Erzin et al. [14] propose a cascaded system for improving recognition rates, in which a novel method, called an adaptive classifier cascade, is developed for selecting the best modalities and their order. The method was applied to identification with audio and video data (face and lip movement), in which separate sets of scores were produced by five classifiers from the same audio and video stream. The method selects classifiers according to the estimated reliability of the modality, this estimation being based on the assumption that a correct speaker model would create a significantly higher likelihood ratio than any other speaker model. The experimental results on a database of 50 persons show that the proposed fusion method outperforms such schemes as product rule and maximum likelihood when selecting the three best out of five individual modalities produced from audio and video data.

The multimodal fusion experiments of many researchers nevertheless suggest that trained classifiers (Neural Networks) [7,15] and simple combination strategies (such as the Weighted Sum rule) [15,16] can achieve good performance in multimodal fusion (although this was not tested in cascaded systems). Such experiments also suggest that performance of a multimodal system can be improved by fusing the most complementary modalities rather than those that perform best [4,17]. We therefore carried out our experiments with trained classifiers and chose the order of the modalities in the cascaded system according to their availability for unobtrusive user authentication. The most unobtrusive and invisible biometric modalities are those based on image processing, such as face biometrics. Another unobtrusive and easily available modality is weight biometrics, but its performance is sufficient only for applications dealing with a small set of users [18]. Voice recognition is also fairly unobtrusive and easily available, so that face plus voice multimodal biometric verification is a suitable combination for the first stages of a verification cascade.

Since neither face nor voice biometrics separately, nor face plus voice fusion, has a sufficiently high performance for many applications, the last stage of the verification cascade should use a reliable modality such as fingerprint or iris biometrics. When designing the verification cascade, it is necessary to investigate two issues. First, what is the difference in performance between a two-stage cascade (which uses both unobtrusive biometric modalities at the first stage) and a three-stage cascade which can accept users based only on one unobtrusive modality in the first stage and on the fusion of two unobtrusive modalities at the next stage. The three-stage cascade can perform user verification unobtrusively more frequently than one that has to wait for two modality samples, but the overall performance of the unobtrusive mode and the complete system might differ between these two configurations. Second, if the explicit stage modality (such as fingerprint or iris) performs much better than the unobtrusive modalities, should the system in the third stage use only the best modality, or should it perform fusion of three modalities anyway? A multimodal biometric system is assumed to have better anti-spoofing capabilities than a single modality system, but the fusion of a well-performing modality with ones that do not perform so well can degrade system performance.

We present here the results of experiments to compare the performances of various configurations of cascaded systems employing different fusion methods. The requirements for a biometric system depend on the application. In case of a parallel multimodal system (which uses all biometric modalities at once), different ratios between the False Acceptance Rate and False Rejection Rate can be selected: high-security applications require a low FAR and have to accept a higher FRR, whereas low-security applications require a low FRR and have to accept a higher FAR. For a cascaded system, applications have a trade-off between the system's FAR at all stages, FRR after the unobtrusive stage and FRR after the last stage. We evaluated empirically how different system configurations and fusion methods affect these parameters.

The main contributions of this paper are: (1) introduction of the idea of cascading unobtrusive multimodal biometrics with a more reliable biometric modality that requires explicit interaction, in order to increase both user-friendliness and system security; (2) demonstration of the feasibility of the method by means of experiments; and (3) comparison of the performances of different system configurations using a database of 150 persons collected in the BioSec project [19].

Since no large multimodal biometric databases exist, the current practice when evaluating the performance of new fusion methods is to use fairly small databases, e.g., the database size in paper [14] was 50 persons and that in [6] was 75 persons. For experiments with the fusion of face and voice data some larger databases do exist, e.g., data on 295 persons from the XM2VTS database were used in [4]. If fusion experiments require other biometric modalities, researchers often need to combine data from several databases, in order to create a multimodal set of scores for each "virtual" person by taking data of one modality from one real person and data of another modality from another real person. Such databases consisting of "virtual" or "chimeric" persons were used in [11,15], for example, and the same method was also used by Snelick et al., who performed experiments on a database of 972 persons [8]. It has been shown recently, however, that using "virtual" persons for fusion purposes cannot appropriately replace a dataset of real users [20]. Our experiments were performed on a reasonably large database of 150 real persons, i.e., the data for all four biometric modalities belong to the same person in each case.

The paper is organised as follows. A short overview of the system and the experiments is presented in Section 2. The database, the experimental protocol and the performances of the individual modalities are presented in Section 3, the results of the experiments are given in Section 4 and discussed in Section 5, and finally the conclusions are presented in Section 6.

2. Overview of the proposed method

In order to increase security levels in a user-friendly way, we propose to perform user authentication by means of unobtrusive biometrics first and to require explicit user authentication only if the unobtrusive stage fails. For an "office door opening" application, explicit authentication is needed immediately after the unobtrusive stage fails, whereas in the case of mobile devices and desktop computers explicit authentication should be required only when an application-dependent security risk arises (see Figs. 1 and 2). One example of a security risk might be predefined timeout during which several attempts at unobtrusive user verification have failed. Such a timeout can be short if the user is working with sensitive data and longer if the user is

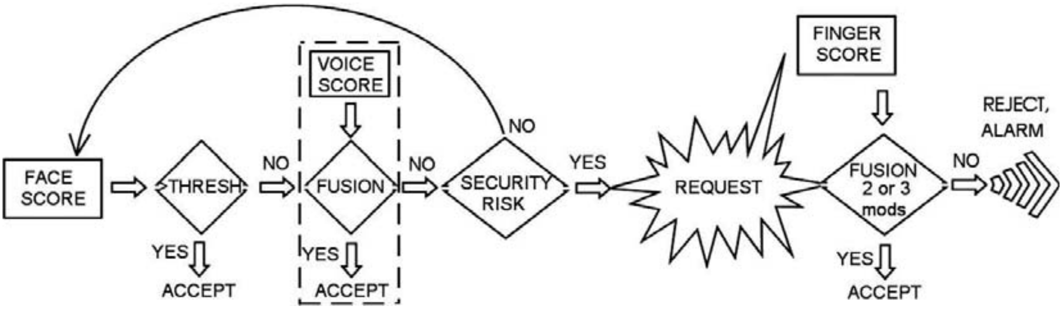


Fig. 1. Operation of a "three-stage" cascaded system (fusion with voice can be skipped if no voice data are available).

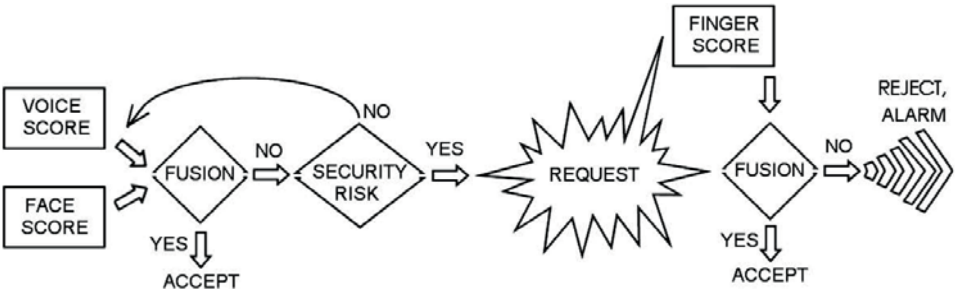


Fig. 2. Operation of a "two-stage" cascaded system.

just drawing diagrams, because a diagram application does not usually present any high-security risks. Another example of security risk can be the recognition of certain contexts, e.g., the opening of a banking application (especially the start of a money transfer) or the beginning of copying sensitive data. The security risk for a mobile device can be considered to be lower if the device is in the user's home and higher in an unknown place.

Security level depends on False Acceptance Rate (FAR) of an authentication method, while user annoyance depends on False Rejection Rate (FRR), and applications have a trade-off between these two types of errors. In a cascaded system false user rejection (in a sense of denial of access to an application) happens only at the last stage, because user rejection by unobtrusive stage causes just a request for a biometric sample, see Figs. 1 and 2. (Number of requests for biometric samples also affects user annoyance, but probably not as much as access denial.) False user acceptance can happen both at the unobtrusive stage and at the last stage ("accept" arrows in Figs. 1 and 2), but from the point of view of security it does not matter at which stage erroneous user acceptance happens, provided that the overall system FAR is kept within the desired limits. We propose that applications set the target FAR according to their desired security levels, and the system training aims at finding optimal parameters in order to keep the overall FAR close to the target FAR and at the same time to minimise both overall FRR and FRR of the unobtrusive stage.

The face modality is the most suitable for unobtrusive user authentication with many devices and in many usage situations. Voice modality is also a natural means of authentication for computer users or users of mobile devices, since most users talk in the presence of their devices from time to time, and since a speech interface is becoming more and more common nowadays. Accordingly, a cascaded system, aiming at as unobtrusive authentication of users as possible, should first try to verify the user by one modality and then combine the two least obtrusive modalities (face and voice), and only after that, if needed, should it ask users to perform explicit authentication by more reliable means such as the iris or fingerprint modality. In order to be as user-friendly as possible, the system should require only one modality for explicit authentication at the last stage. Thus, if no user's voice data are available, the system should perform authentication using only face data and then move on to a reliable explicit modality if necessary.

Another way to perform unobtrusive user verification is not to try to verify the user from face data but to use face and voice data simultaneously (see Fig. 2). Depending on the application requirements and on how frequently the user's speech is expected to be used, it might be feasible to employ a "two-stage" cascade (which requires both face and voice data at the first stage) instead of "three-stage" system. Since the performance of a parallel multimodal system (which uses face and voice data simultaneously for authentication, for example) is usually better than that achievable with the face modality only, such a "two-stage" cascade should require explicit authentication for smaller number of cases, while a "three-stage" cascade can operate without voice data at all and should thus be suitable for the verification of silent users.

At the last, most reliable, stage in the cascade either the best single modality or a fusion of all available scores can be used, depending on the performance of the best modality and its anti-spoofing capabilities. In our tests we compared system performance at the last stage in two configurations: using a fusion of all the available scores for making the final decision and basing the final decision on the best modality score alone.

3. Database and experimental protocol

3.1. The multimodal database and experimental protocol for the individual modalities

The multimodal database, collected in the course of the BioSec project, contains data on 200 persons with respect to four biometric modalities: face, voice, fingerprint and iris [19]. The data were collected in two sessions, after which the project partners processed the data and produced similarity scores for individual modalities following the Evaluation Protocol designed by the Performance Evaluation Board of the BioSec project. Database collection and the evaluation protocol were designed in a user-friendly fashion, so that only one biometric sample for each user was taken to produce a template for training the corresponding individual modality. By the time of our experiments we had single modality scores for the following data:

- Four face samples taken in each data collection session.
- Four iris images from the right eye taken in each session.
- Four fingerprint images from the right index finger taken in each session.
- Four voice utterances in English taken with a web cam microphone in each. session

With the exception of the fingerprint and iris scores, scores of all the other individual modalities were produced using the following protocol: first, the single modality algorithms were trained and tuned on the basis of a development set consisting of 50 users (the first and last 25 users in the database). Second, the single modality algorithms produced scores for the evaluation set, which consisted of the remaining 150 users. The fingerprint and iris scores were produced by the recognition methods developed earlier, without any optimisation of the parameters. Only the scores for the evaluation set (150 users) were used in the multimodal experiments, so that the scores for all four individual modalities corresponded to the same real person.

The similarity scores for the individual modalities were produced by the following BioSec partners: the face scores were provided by the Aristotle University of Thessaloniki, Greece, the voice scores by the Universidad Politecnica de Madrid, Spain, the iris scores by Naukowa i Akademicka Siec Komputerowa, Poland, and the fingerprint scores by Alma Mater Studiorum – Università Di Bologna, Italy. In all cases the scores were produced according to the following experimental protocol:

- *Genuine scores*: each of the samples from the first session matched with the four samples from the second session.
- *Impostor scores*: the first sample from the first session matched with the first sample from every other user in the first session, avoiding symmetrical matches.

This experimental protocol presented difficulties for individual modalities: the requirement that only one biometric sample for each user should be used at the enrolment stage did not allow any discarding of poor quality samples. This protocol was especially difficult to follow for the iris modality because occlusion of the iris images (due to eye blinking) happens frequently and the usual practice in iris recognition would be to use several images at the enrolment stage for better detection of iris occlusions and eye rotations. Consequently, the poorest samples were excluded from our experiments and the multimodal experiments were carried out with 1100 genuine scores and 2350 impostor scores for each modality. Due to the fairly difficult experimental protocol, the performances of the single modalities were not very good, see Fig. 3. FAR and FRR for the single modalities were calculated in the following way: first we set the upper limit for a target False Acceptance Rate, and then we selected a threshold in such a way that the FAR for the training set (half of the data) would be below the target FAR, but not less than 70% of it. Using this threshold, we calculated the errors for the test set (the other half of the data). After that the training and test sets were swapped round, the training and testing repeated and the error rates averaged between the two trials. The protocol and training/test data sets were exactly the same as in the multimodal experiments, for details see Section 3.2.

The success of multimodal fusion largely depends on how mutually complementary the single modalities are (whether they make errors for the same person or for different persons). As seen in Fig. 4, the genuine and impostor scores for the voice and face modalities overlap considerably, which means that the face and voice modalities don't complement each other very well in our database. This presents additional difficulties for fusion. On the other hand, such overlapping is probably closer to a real-life situation, when a large number of people in a room can present more challenges for both audio and image processing by increasing the noise level and producing a more scattered and occluded picture for image processing.

3.2. Algorithms and experimental protocol for the cascade experiments

For each genuine and each impostor transaction we had set of four scores, obtained by comparison of a biometric

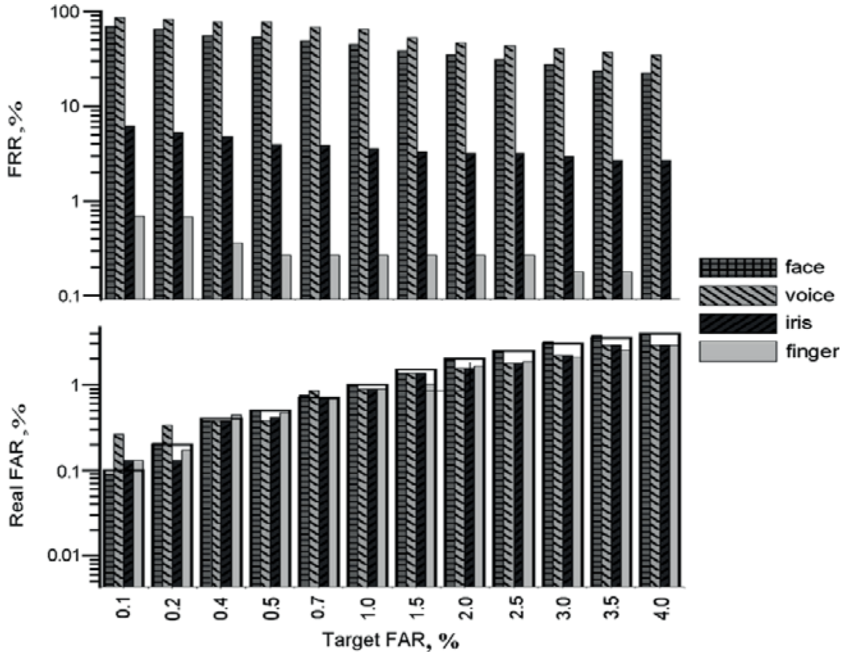


Fig. 3. Performances of single modalities at different target False Acceptance Rates, %.

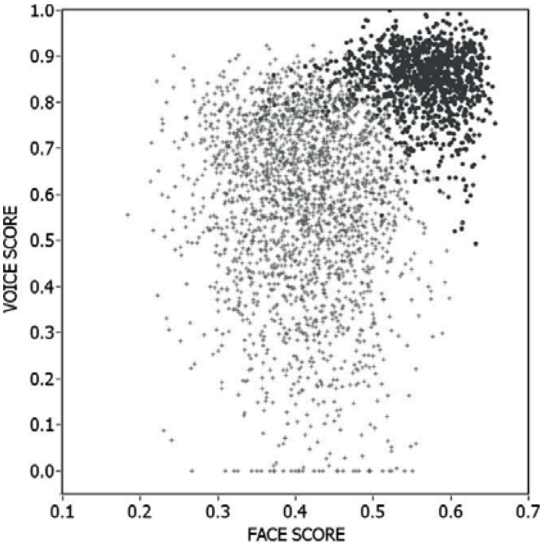


Fig. 4. Overlap of clients and impostor scores in the face (horizontal axis) and voice (vertical axis) modalities. Black circles denote clients, grey crosses denote impostors.

sample of one user against a sample from the same (genuine score) or different (impostor score) user, so that the scores for each single modality belonged to the same real person. The data were divided equally into “Set 1” and “Set 2”, and the experiments were performed by first training each algorithm on “Set 1” and tested it on “Set 2” and then training it on “Set 2” and testing it on “Set 1”, after which the test results were averaged.

The training and testing was performed for “three-stage” cascade in a following way: first, we set the upper limit for a target FAR (False Acceptance Rate) for the system and a lower limit which was selected to be 70% of the upper limit. Next, a threshold for the first stage of the cascade (face modality only) was selected so that the FAR for the training set would be within the desired limits. The persons accepted by the first stage were excluded from the training of the next stages. The second stage was then trained on the remaining training data in such a way that the FAR after the two stages would fall within the limits. (The second stage was skipped when simulating the case of “silent users”, i.e., the system operation without voice data.) The persons accepted at these unobtrusive stages were excluded from the training of the last stage, which in turn was trained so that the FAR for the whole cascade would fall within the desired limits. The reason for excluding persons accepted by the previous stages from the training of the next stage was twofold: first, training is faster with less data, which is important for real-life applications, and second, the performance of the cascaded system in the case of training on a whole data set, which we also tested, was similar to that achieved with the smaller amount of data (sometimes slightly better, sometimes slightly worse).

The upper limits for the target FAR were chosen in order to test the suitability of the cascade for both high- and low-security applications. For higher security requirements we selected the following group of fairly low target FAR values: 0.1%, 0.2%, 0.4%, 0.5%, 0.7% and 1%, while for lower security applications we selected fairly high target FAR values: 1.5%, 2%, 2.5%, 3%, 3.5% and 4%.

Training and testing of the "two-stage" cascade was performed using the same upper and lower limits for the target FAR values, in the following way: first the unobtrusive stage (face plus voice) was trained to achieve a FAR within the desired limits, after which the second stage was trained on the remaining data to achieve a FAR for the whole cascaded system that was within the desired limits.

The experiments with score-level fusion were performed using the following algorithms: Weighted Sum rule, SVM (Support Vector Machines) and MLP (Multi-Layer Perceptron). In the case of the "three-stage" cascade fusion was performed at both the second and third stages using the same algorithm (see Tables 1 and 2 in the next section). The TORCH library of machine-learning algorithms [21] was used for the experiments with SVM and MLP, employing the default TORCH settings (number of hidden nodes 30, one hidden layer with Tahn activation functions for MLP, Gaussian kernel for SVM), because we had observed in our previous numerous experiments with TORCH that although better performance can be achieved with other settings, the improvement is not very significant. Also, the algorithms do not always converge in non-default configurations.

4. Experimental results

This section presents the system performance results achieved with different fusion methods and in different configurations, and compares the use of iris and fingerprint modalities at the last stage. The configurations tested for the "three-stage" cascade and their legends in the graphs are presented in Table 1 and those for the "two-stage" cascade in Table 2. In the case of silent users (no voice data available) the "three-stage" cascade in practice performs the fusion in two stages, but the term "three-stage" will nevertheless be used below in order to differentiate this system configuration from the "two-stage" cascade, which always performs fusion of both unobtrusive modalities at the first stage. In all the configurations the system was given only one chance to perform user verification by means of unobtrusive modalities.

Thus we tested 10 configurations of the "three-stage" cascade and six of the "two-stage" cascade for each explicit modality (iris or fingerprint), and also evaluated the performance of a parallel multimodal system (which uses the scores for three modalities simultaneously) for the iris and fingerprint modalities. Since our main goal was to investigate the possibilities for increasing system unobtrusiveness without reducing security, we will not present the performance results for the parallel system in graph form. It should be noted, however, that our experimental results are encouraging, in that the difference between the performance of the parallel system and that of the corresponding cascaded system was not statistically significant, although

Table 1
Configurations of the "three-stage" cascade tested in our experiments, and their legends in the graphs

"First Stage"	"Second Stage" and its *legend* in graph 4	"Third Stage" and its *legend* in graphs 5 and 6 for "talkative users" and in graphs 7 and 8 for "silent users" (second stage skipped due to absence of voice data)
Face recognition alone	Face-voice fusion, Weighted Sum – *sum3*	Face-voice-explicit modality (either iris or fingerprint) fusion, Weighted Sum – *sum3* Explicit modality (either iris or fingerprint) alone – *not shown in the graphs*
	Face-voice fusion, MLP –*MLP3*	Face-voice-explicit modality (either iris or fingerprint) fusion, MLP – *MLP3* Explicit modality (either iris or fingerprint) alone – *not shown in the graphs*
	Face-voice fusion, SVM – *SVM3*	Face-voice-explicit modality (either iris or fingerprint) fusion, SVM – *SVM3* Explicit modality (either iris or fingerprint) alone – *SVM3LI* or *SVM3LF*
	Skipped if no voice data are available	Face-explicit modality (either iris or fingerprint) fusion, Weighted Sum – *sum* Face-explicit modality (either iris or fingerprint) fusion, MLP - *MLP* Face-explicit modality (either iris or fingerprint) fusion, SVM - *SVM* Explicit modality (either iris or fingerprint) alone – *Iris* or *Finger*

Table 2
Configurations of the "two-stage" cascade tested in our experiments

"First Stage" and its *legend*	"Second Stage" and its *legend* in graphs 5 and 6
Face-voice fusion, Weighted Sum – *sum2*	Face-voice-explicit modality (either iris or fingerprint) fusion, Weighted Sum – *sum2* Explicit modality (either iris or fingerprint) alone – *not shown in the graphs*
Face-voice fusion, MLP – *MLP2*	Face-voice-explicit modality (either iris or fingerprint) fusion, MLP – *MLP2* Explicit modality (either iris or fingerprint) alone – *not shown in the graphs*
Face-voice fusion, SVM – *SVM2*	Face-Voice-Explicit Modality (either iris or fingerprint) fusion, SVM – *SVM2* Explicit modality (either iris or fingerprint) alone – *SVM2LI* or *SVM2LF*

for almost the whole range of target FARs the performance of the parallel system was slightly better. Thus, replacement of the parallel system architecture with a cascaded one is feasible.

The experimental results are presented in the figures below. The performances of the unobtrusive mode in various system configurations for different target False Acceptance Rates are presented in Fig. 5. For the "three-stage" cascade we present both system performances after the first stage, face only, and after the second stage, face plus voice fusion, while for the "two-stage" cascade we present the results of face plus voice fusion at the first stage.

System performance after the last stage, i.e., after explicit interaction with the best modality (fingerprint or iris), is presented in Figs. 6 and 7. Where the last stage in the cascade used only the score from the best (explicit) modality, only graphs for the best case (SVM fusion in the unobtrusive stage) are included in the figures and not graphs for MLP and Weighted Sum fusion at the unobtrusive stage. The performance of a cascade built up of the face, voice and fingerprint modalities is presented in Fig. 6 and that of a cascade built up of the face, voice and iris modalities in Fig. 7.

System performance when no voice data were available during operation of the "three-stage" cascade, which actually turns into two-stage cascade in this case, is presented in Figs. 8 and 9: the first stage is the face modality only, the second stage is skipped and the last stage is either a fusion of face plus iris data or face plus fingerprint data, or explicit modality alone. The figures show that although the voice modality alone performs less well than any other modality, its use in a multimodal system improves performance relative to a "silent mode".

5. Discussion

The proposed method of cascading unobtrusive biometrics with a more reliable biometric modality which requires explicit interaction can serve two goals. First, it is usually beneficial if user verification happens unobtrusively, so that a computer login or an "open a door application" can be performed hands-free, without taking up the user's time or effort. Second, frequent user authentication can increase the security level of many computer systems, but explicit authentication is too disturbing and normally takes place only during login. Current state-of-the-art biometrics does not provide means for unobtrusive secure authentication, and this problem cannot be expected to be solved completely in the near future. Many office buildings and private homes, for example, have already security cameras installed which could be used for unobtrusive verification, but the poor quality of the images and voice samples presents additional challenges. One possible solution to this problem would be a cascaded system which first attempts

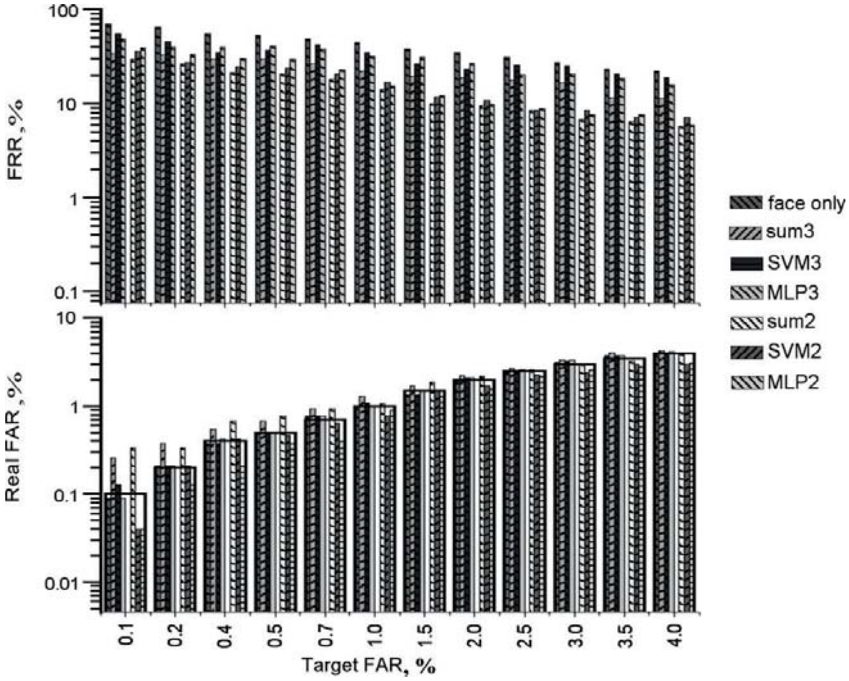


Fig. 5. Performance of the unobtrusive mode in various system configurations at different target False Acceptance Rates, %. Black boundary in the FAR graph indicates the target FAR.

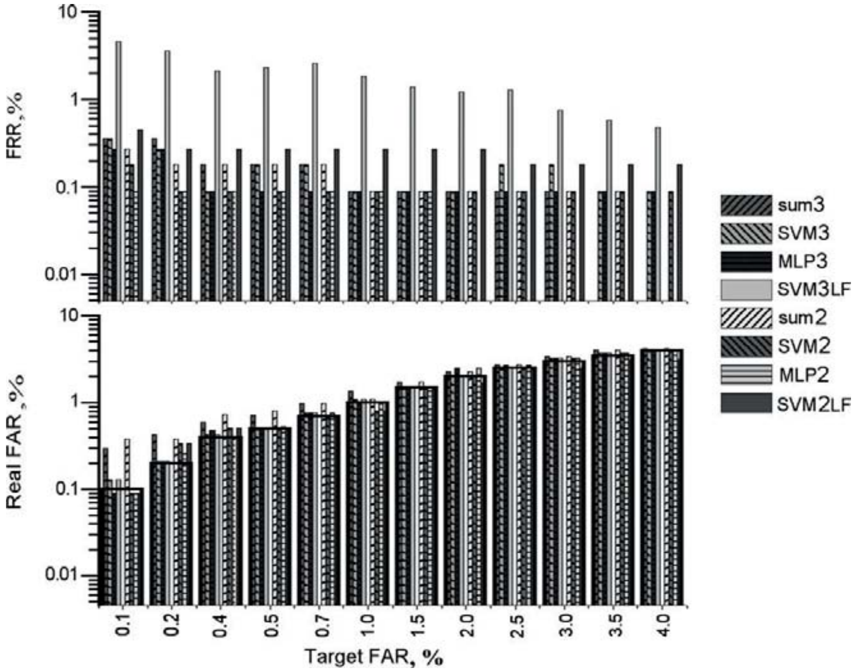


Fig. 6. Performance of a face, voice and fingerprint cascade in different system configurations. Sum3, SVM3 and MLP3 denote a "three-stage" cascade with the last stage based on fusion of all three modalities; Sum2, SVM2 and MLP2 denote a "two-stage" cascade with the last stage based on fusion of all three modalities; and SVM3LF and SVM2LF denote "three-stage" and "two-stage" cascades with the last stage based on the fingerprint score only. The black boundary in the FAR graph indicates the target FAR.

to perform user verification unobtrusively and requires explicit authentication only if the unobtrusive authentication fails and decision is needed immediately.

The main idea of such a cascaded system is that the system should perform unobtrusive user verification frequently and keep a history of it. If the recent history of user verification is not successful, this situation is considered risky, and the system can ask the user to provide a biometric sample of more reliable modality. The notion of "recent history" is application-dependent, so that if a user starts a sensitive application, for example, valid unobtrusive verification could be necessary during last minute(s), whereas a longer time period could be set for other applications. In this work, however, the cascade was allowed only one attempt of unobtrusive verification before using an explicit modality.

The present experiments were carried out with a cascaded system which first performs user verification by unobtrusive face and voice biometrics; and then uses a more precise modality, requiring explicit interaction, at the last stage if necessary. We experimented with two modalities for the last stage of the system, iris and fingerprint (the recognition rate of the fingerprint modality was much higher than that of the iris modality, because our data contained many images with iris occlusion), and with three fusion methods (Multi-Layer Perceptron, Support Vector Machines and Weighted Sum), and compared the behaviour of the system in the various configurations.

A cascaded system can be designed in many ways. We selected the order of modalities according to their availability for unobtrusive user verification: face, voice and a modality which requires explicit interaction (fingerprint or iris). After fixing the order of modalities, a choice has to be made between a "two-stage" cascade (which uses both face and voice data at the first stage) and a "three-stage cascade" (which first attempts to verify the user only by face, adds the voice modality if face verification fails and uses either all three modalities or face and an explicit modality at the last stage). Experimental comparison of the "two-stage" and "three-stage" cascade configurations showed that for a group of fairly high target FARs (higher than 1.5%) the False Rejection Rate in the last stage of the "two-stage" cascade was significantly lower than that for the "three-stage" cascade (at the confidence level over 95%) when iris recognition and fusion by the Weighted Sum method were used, whereas in the case of Weighted Sum fusion for fingerprint recognition there was no statistically significant difference in performance between the "two-stage" and "three-stage" cascade configurations. The results of MLP fusion suggest better performance in the case of the "two-stage" cascade, but this was not always statistically significant. (Unlike the Weighted Sum,

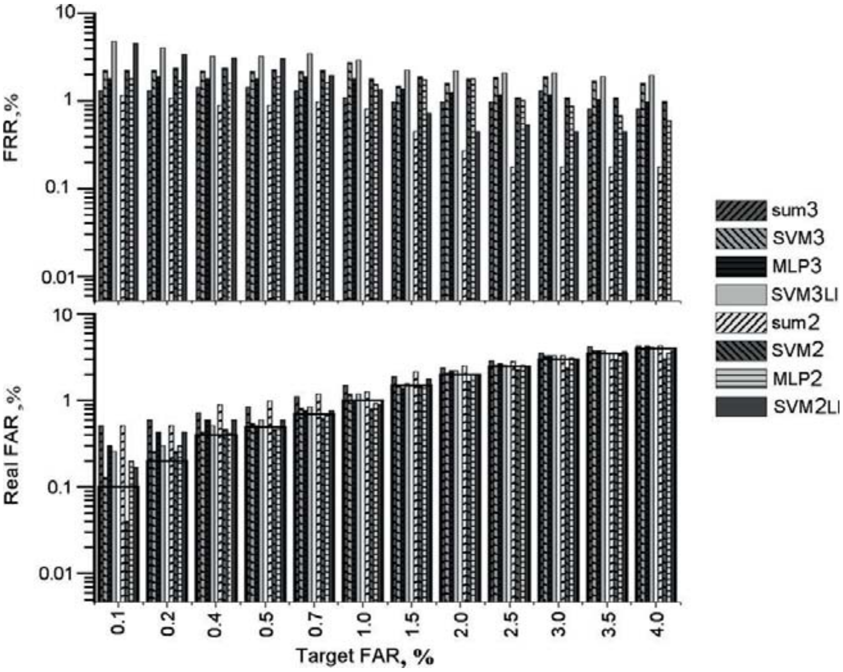


Fig. 7. Performance of a face, voice and iris cascade in different system configurations. Sum3, SVM3 and MLP3 denote a "three-stage" cascade with the last stage based on fusion of all three modalities; Sum2, SVM2 and MLP2 denote a "two-stage" cascade with the last stage based on fusion of all three modalities; and SVM3LI and SVM2LI denote "three-stage" and "two-stage" cascades with the last stage based on the iris score only. The black boundary in the FAR graph indicates the target FAR.

the difference between the two configurations was statistically more significant for the fingerprint modality in the case of MLP fusion). The difference in performance between the "two-stage" and "three-stage" cascade configurations was not statistically significant in the case of SVM fusion (for either iris or fingerprint recognition).

With low target FARs the difference in performance between the "two-stage" and "three-stage" cascade configurations was not statistically significant for any of the fusion methods used. We suggest that the choice between these two configurations should depend on the application and its user interface, namely how frequently a user voice sample is available. If a target FAR of less than 0.4% is required, for example, unobtrusive verification with a "three-stage" cascade will fail in 34.9% ± 3.6% of cases (at the 95% confidence level), whereas with a "two-stage" cascade it will fail in 24.7% ± 3% (at the 95% confidence level) of cases. However, a "three-stage" cascade will verify 44.5% ± 4% (at the 95% confidence level) of cases by means of only the first modality, which is not a small number. (The above results were obtained with SVM fusion, but the other classifiers show similar trends). Consequently, although the performance of the unobtrusive mode in the "two-stage" cascade is significantly better, a "three-stage" cascade can be used in cases when voice data are expected to be available infrequently or not available at all.

Although the voice modality has the highest error rates (EER about 13%), the performance of the "three-stage" cascade in a "silent mode" (with no voice data available) was significantly poorer than in a "talkative mode" when combined with iris recognition and non-significantly poorer with fingerprint recognition (see Fig. 3 for the performance of the fingerprint and iris modalities alone, and Figs. 8 and 9 for the performances of the "silent mode" with fingerprint or iris). Nevertheless, the system performance remains within acceptable limits, as the performance of the "silent mode" was better (at the confidence level over 90%) than that of the best modality alone with iris recognition for a low target FAR (in a range of less than 0.1%–less than 1%), and the performance of the "silent mode" was also better in the case of the fingerprint modality, although the difference was not statistically significant. For a high target FAR, the performance of the "silent mode" of the cascade with iris recognition was similar to that achieved using the iris modality alone, while the performance of the "silent mode" with fingerprint recognition was poorer than that achieved using only fingerprint recognition for user verification, although again the difference was not statistically significant. Thus even in a "silent mode" the cascade is an advantageous system configuration because it requires less explicit interaction than a system that does not use unobtrusive modalities.

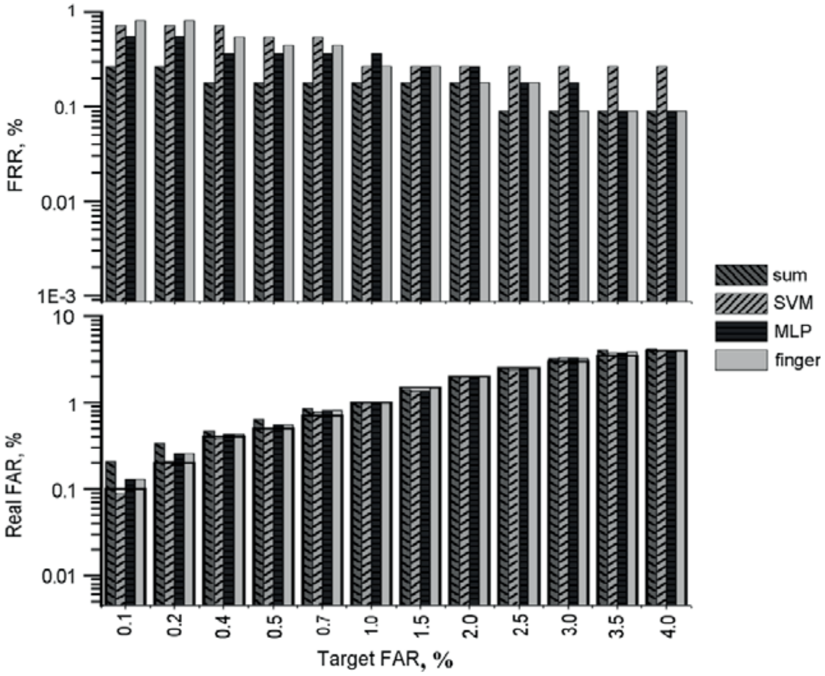


Fig. 8. Performance of the "three-stage" cascade for silent users (no voice data available), using the face and fingerprint modalities. The first stage is the face modality only and the second stage either a fusion of the face and fingerprint scores or the fingerprint modality only. The black boundary in the FAR graph indicates the target FAR.

The next choice to be made in a cascaded system is how to design the last stage: to use all three available modalities together or to use only the best modality for users, not accepted by the previous stages. This choice depends on the performance of the best modality and its anti-spoofing protection. In our tests the performance of the multimodal system at the last stage was very similar to that achieved using only the best modality in the case of Weighted Sum fusion, whereas in the case of SVM or MLP fusion the performance of the multimodal system was better than if only the best modality was used. For low target FARs the confidence level of this finding was over 95% with both iris and fingerprint recognition in the "three-stage" cascade, but only 85–99% for the iris modality and 75–90% for the fingerprint modality in the "two-stage" cascade. With high target FARs this finding was statistically more significant for the fingerprint than for the iris modality in the "three-stage" cascade and the reverse in the "two-stage" cascade. Although use of only the best modality at the last stage improved the performance of the "two-stage" cascade with high target FARs in some cases, this improvement was insignificant, and use of only the best modality at the last stage in the "silent mode" also degraded the recognition rates. This is an interesting result, because the performance of the fingerprint modality was significantly better than that of the iris modality (see Fig. 3), so that one could expect that its use in the last stage should require a different system configuration than use of the iris modality. The experiments did not confirm this expectation, however. We can conclude from this result that it is feasible from the point of view of both performance and anti-spoofing to use fusion of all three modalities in the last stage of a high-security system even if the best modality is significantly better than the others. Using only the best modality in the last stage is feasible only for low-security applications, because performance of this system configuration with a high target FAR is similar to that of multimodal fusion, but implementation without fusion is easier and requires less system training.

Comparison between the classifiers used in our experiments suggests SVM as the first choice for high-security applications, as it was the only fusion method capable of keeping the system FAR within the desired limits in most cases. Moreover, the main reason for exceeding the target FAR in a "three-stage" cascade with SVM in the system was often that target had been exceeded by the first stage (face only), while fusion had not increased the FAR any more. The Weighted Sum rule turned out to be the least capable of keeping the FAR below the target. On the other hand, if security requirements are not high (keeping the FAR within the desired limits is not crucial), the Weighted

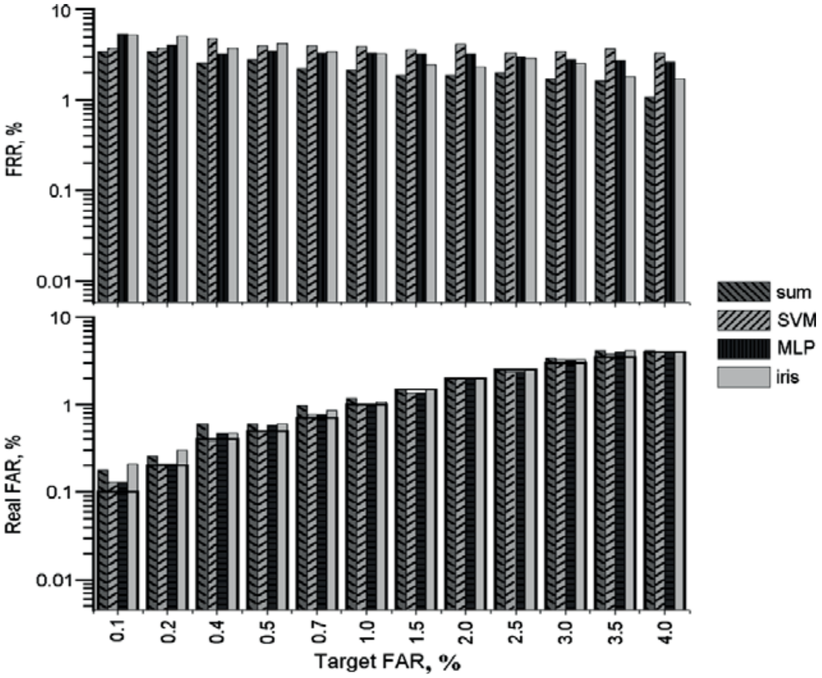


Fig. 9. Performance of the "three-stage" cascade for silent users (no voice data available), using the face and iris modalities. The first stage is the face modality only and the second stage either a fusion of the face and iris scores or the iris modality only. The black boundary in the FAR graph indicates the target FAR.

Sum rule may be a suitable method, because in most cases it produced the lowest False Rejection Rates. (The difference between the Weighted Sum and SVM performances was statistically significant from the point of view of both FAR and FRR.)

MLP appeared to produce the least predictable results. In some cases it was capable of keeping the FAR within the desired limits (mainly in a "two-stage" configuration and a "three-stage" configuration with a high target FAR), but in many cases the FAR exceeded the desired limits with MLP. Although the difference in performance between MLP and SVM was not statistically significant in many cases, the unstable behaviour and longer training times associated with MLP suggest that SVM would be a better choice for high-security applications.

The comparison between the "two-stage" and "three-stage" configurations showed that the ability to keep the FAR within the desired limits was higher in the "two-stage" system, but this difference was not statistically significant in most cases. The difference between Weighted Sum and SVM fusion in this respect was nevertheless statistically significant in both the "two-stage" and "three-stage" configurations, so that the ability of the system to keep the FAR within the desired limits depends more on classifier selection than on the number of stages.

It is worth noting that the BioSec database of face and voice scores is fairly difficult for fusion due to high rate of overlapping between the scores for genuine users and impostors (see Fig. 4). We believe that such overlapping reflects the real-life situation, in that environments with many people gathered together and moving about usually have both a high audio noise level and a scattered background, and this makes both audio and image processing more difficult. The protocol for the training/testing of the single modalities in the BioSec project was also fairly difficult, because it required producing a template from a single biometric sample only, which was especially unsuitable for iris recognition. On the other hand, the protocol is very easy for users to follow, because they do not need to interact with a biometric sensor again even if the data are faulty.

6. Conclusions

This paper proposes cascading unobtrusive user verification with a more reliable biometric modality that requires user cooperation, with the aim of reducing user effort and increasing the security of the system. Numerous experiments were carried out in order to study how different system configurations and fusion methods affect the performance of the unobtrusive stage and of the overall system. The unobtrusive biometric modalities chosen for examination were face and speaker recognition, and the more reliable biometric modalities tested were fingerprint and iris recognition.

Since our main goal was to investigate the possibilities for increasing system unobtrusiveness without reducing security, we do not present here a detailed comparison of the performance of a cascaded system with that of a parallel system (which uses all three modalities simultaneously and thus always requires explicit user interaction). In brief, the performance of the parallel system was usually better than that of the cascaded system, but the difference was not statistically significant.

The comparison between the performance of a cascaded system and that of a system which verifies users only by means of an explicit modality shows that in most cases (especially with high-security requirements) the performance of the cascaded system was significantly better. The comparisons between the fusion methods tested (SVM, MLP and Weighted Sum) and between using fusion and using only the best modality in the last stage suggest that the best method for high-security requirements is SVM fusion at all stages, while simple system configurations (such as Weighted Sum fusion at all stages, or even using only the best modality at the last stage) are better suited for low-security requirements, which is also a good result from the point of view of ease of deployment.

The comparison between a “three-stage” cascade and a “two-stage” cascade suggests that the performance of the last stage of a “two-stage” cascade is significantly better with a high target FAR, whereas the difference was not statistically significant with a low target FAR. Unobtrusive verification is significantly better in a “two-stage” cascade but nevertheless it is still feasible using a “three-stage” cascade if a speech input is expected only infrequently or not at all. The choice between a “two-stage” and a “three-stage” system should depend on how frequently user voice data can be expected. Although the performance of a cascade in “silent mode” was poorer than that of any other cascade configuration, it was not significantly different from the performance of the best (explicit) modality alone.

Despite the fairly poor performance (EER above 10%) and high overlap of errors of unobtrusive modalities, the proposed method has proved to be able to satisfy both high- and low-security requirements. Even if the system requirement was a very low False Acceptance Rate of less than 0.1%, the “two-stage” cascade was able to verify the users unobtrusively (by face and voice fusion) in $64.7\% \pm 2.4\%$ of cases, while the False Rejection Rate of a complete system was $0.18\% \pm 0.12\%$. In lower security settings (say an acceptable FAR of 1.5%) the False Rejection Rate achieved by the unobtrusive mode was $12\% \pm 1\%$; while that of the complete system was $0.09\% \pm 0.09\%$ (almost nobody would be rejected erroneously).

To conclude, the results presented here represent the first study of a multimodal cascaded biometric verification system, proposed with the aim of reducing user effort, and address the pros and cons of different configurations. The experimental results confirm the feasibility of the proposed method from the point of view of both its overall performance in relation to high- and low-security requirements and its ability to reduce user effort.

References

[1] A. Miller, PDA security concerns, Network Security 2004 (7) (2004) 8–10.

[2] A. Jain, A. Ross, Sh. Pankanti, Biometrics: a tool for information security, IEEE Transactions on Information Forensics and Security 1 (2) (2006) 125–143.

[3] N. Poh, S. Bengio, Towards predicting optimal fusion candidates: a case study on biometric authentication Tasks, in: the 1st International Machine Learning and Multimodal Interaction Workshop 2004 (MLMI’04), LNCS, vol. 3361, pp. 159–172.

[4] F. Roli, J. Kittler, G. Fumera, D. Muntoni, An experimental comparison of classifier fusion rules for multimodal personal identity verification systems, multiple classifier systems, in: Third International Workshop, MCS 2002, Cagliari, Italy, June 2002, pp. 252–261.

[5] A. Jain, A. Ross, Learning user-specific parameters in a multibiometric system, in: Proceedings of the International Conference on Image Processing (ICIP), Rochester, NY, September 2002, pp. 57–60.

[6] J. Fierrez-Aguilar, D. Garcia-Romero, J. Ortega-Garcia, J. Gonzalez-Rodriguez, Adapted user-dependent multimodal biometric authentication exploiting general information, Pattern Recognition Letters 26 (16) (2005).

[7] S. Bengio, C. Marcel, S. Marcel, J. Mariethoz, Confidence measures for multimodal identity verification, Information Fusion 3 (4) (2002) 267–276.

[8] R. Snelick, U. Uludag, A. Mink, M. Indovina, A. Jain, Large scale evaluation of multimodal biometric authentication using state-of-the-art systems, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (3) (2005) 450–455.

[9] J. Koreman, A.C. Morris, D. Wu, S. Jassim, H. Sellaheva, J. Ehlers, J. Chollet, G. Aversano, H. Bredin, S. Garcia-Salisetti, L. Allano, B. Ly Van, B. Dorizzi, Multi-modal biometric authentication on the SecurePhone PDA, in: Second Workshop on Multimodal User Authentication MMUA 2006, Toulouse, France, May 2006.

[10] Li Huang, Learning with cascade for separation of non-convex manifolds, in: IEEE Workshop on Face Processing in Video, Washington, USA, June 2004.

[11] L. Hong, A. Jain, Integrating faces and fingerprints for personal identification, IEEE Transactions on Pattern Analysis and Machine Intelligence 20 (12) (1998) 1295–1307.

[12] A. Jain, A. Ross, S. Prabhakar, An introduction to biometric recognition, invited paper, IEEE Transactions on Circuits and Systems for Video Technology 14 (1) (2004).

[13] K. Takahashi, M. Mimira, Y. Isobe, Y. Seto, A secure and user-friendly multimodal biometric system, in: A. Jain, N. Ratha (Eds.), Biometric Technology for Human Identification, Proceedings of SPIE, vol. 5404, August 2004.

[14] E. Erzin, Y. Yemez, A.M. Tekalp, Multimodal speaker identification using an adaptive classifier cascade based on modality reliability, IEEE Transactions on Multi-media 7 (5) (2005) 840–852.

[15] Y. Wang, T. Tan, A. Jain, Combining face and iris biometrics for identity verification, in: Proceedings of the 4th International Conference on Audio- and Video-Based Biometric Person Authentication (AVBPA), Guildford, UK, June 2003, pp. 805–813.

[16] K. Jain, A. Ross, Multibiometric systems, Communications of the ACM 47 (1) (2004) 34–40.

[17] V. Chatzis, A. Bors, I. Pitas, Multimodal decision-level fusion for person authentication, IEEE Transactions on Systems, Man, and Cybernetics 29 (6) (1999) 674–680.

[18] H. Ailisto, M. Lindholm, S.-M. Mäkelä, E. Vildjiounaite, Unobtrusive user identification with light biometrics, in: Proceedings of the Third Nordic Conference on Human–Computer Interaction, ACM Press, Tampere, Finland, October 2004, pp. 327–330.

[19] J. Fierrez-Aguilar, J. Ortega-Garcia, D. Torre-Toledano, J. Gonzalez-Rodriguez, BioSec baseline corpus: a multimodal biometric database, Pattern Recognition 40 (4) (2007) 1382–1392.

[20] N. Poh, S. Bengio, Can chimeric persons be used in multimodal biometric authentication experiments? MLMI (2005).

[21] Available from: <http://www.torch.ch/>.

PUBLICATION II

Unobtrusive dynamic modelling of TV programme preferences in a Finnish household

Multimedia Systems, Vol. 15 (2009), No. 3, pp. 143–157.

PUBLICATION III

Semi-supervised context adaptation: case study of audience excitement recognition

Multimedia Systems, Vol. 8 (2012), Issue 3, pp. 231–250.

PUBLICATION IV

Prediction of interface preferences with a classifier selection approach

Journal on Multimodal User Interfaces,
Vol. 7 (2013), Issue 4, pp. 321–349.

PUBLICATION V

Requirements and software framework for adaptive multimodal affect recognition

Proceedings of the 2009 3rd International Conference on
Affective Computing and Intelligent Interaction (ACII 2009).
Amsterdam, The Netherlands,
10–12 Sept. 2009. IEEE Press. 7 p.

Requirements and Software Framework for Adaptive Multimodal Affect Recognition

Elena Vildjiounaite, Vesa Kyllönen, Olli Vuorinen, Satu-Marja Mäkelä, Tommi Keränen,
Markus Niiranen, Jouni Knuutinen, Johannes Peltola
VTT Technical Research Centre of Finland
Kaitoväylä 1, 90571 Oulu, Finland
firstname.lastname@vtt.fi

Abstract

This work presents a software framework for real time multimodal affect recognition. The framework supports categorical emotional models and simultaneous classification of emotional states along different dimensions. The framework also allows to incorporate diverse approaches to multimodal fusion, proposed by the current state of the art, as well as to adapt to context-dependency of expressing emotions and to different application requirements. The results of using the framework in audio-video based emotion recognition of an audience of different shows (this is a useful information because emotions of co-located people affect each other) confirm the capability of the framework to provide desired functionalities conveniently and demonstrate that use of contextual information increases recognition accuracy.

1. Introduction

Existing works on multimodal affect recognition largely fall into two categories: first, thorough offline studies (the work [1] presents a recent survey on audio-visual affect recognition); second, real-time interactive applications developing an affect recognition method for a particular task (as in the work [2]). We have not found works presenting multimodal fusion software for real time affect recognition, which would allow to adapt on the fly to changes in data availability, environments, users and application tasks (e.g., to take into account that same emotion may be expressed differently if a person is talking to a boss than if he/she is talking to a spouse) and in the same time to utilize diverse methods of increasing recognition accuracy. Some of the listed above functionalities are provided by generic machine learning libraries, but these libraries suit mainly for offline comparison of reasoning methods [3].

Context recognition and adaptation to users and contexts is an actively developing research area, but again adaptation is usually done in an application-specific manner. This research provided methods to recognize diverse situations automatically, for example, to use address book of a phone for distinguishing between calls to a boss and to a spouse; to acquire user location via GPS and services providing coordinates of main points of interest such as museums, concert halls, stadiums etc; other information about user situation (such as a formal dinner with business partners vs. a party with friends) can be acquired from a personal calendar [4]. Consequently, it becomes possible to use context in affect recognition, but the survey [1] stated the need to take into account context of expressing emotions as an important, but rarely addressed issue. Dynamics of emotional states was listed as another important issue.

This work presents a software framework allowing to deal with these and other important issues with a little configuration effort, and the experiments with using the framework for audio-visual recognition of emotions of an audience in different contexts. Emotion recognition of an audience may be useful for interactive installations, for giving a prize of audience preferences, for memory aid tools and also because emotions of surrounding people affect emotions of an individual. For example, liking or dislike of others affect personal mood if a person watches TV in a company [5].

2. Current Trends in Affect Recognition

Approaches to emotion recognition differ, first, in choice of emotional models. Use of categorical models is a more common approach because such models are used by humans in daily life and thus labelling of collected emotional data with categorical models is quite natural. Categories used by different researches include some (or none) of basic Ekmanian emotions (joy, sadness, fear, anger, surprise and disgust) and/ or some other categories such as frustration [6] or boredom [7]. Choice and number of categories depend on the application goal, for example, the work [8] aims at distinguishing between two emotional categories only (fear and neutral) for surveillance purposes. Another approach is to use dimensional models, such as PAD (Pleasure/ Arousal/ Dominance) model, but labelling of emotional data with dimensional models is more difficult and thus either non-trainable fusion rules are used [2] or special training of annotators is required before labelling [1]. Labelling can be also simplified to classification into selected sectors (e.g., positive/ negative, low/ mid/ high) at the expense of information loss [1].

978-1-4244-4799-2/09/$25.00 ©2009 IEEE

Second, approaches to multimodal emotion recognition differ in choice of fusion methods: fusion can be performed on decision level [9], that is, each modality outputs a “final” category and these outputs are fused, for example, by voting. Majority of works on multimodal fusion uses lower-level fusion methods, so that each modality outputs one of modality-dependent classes and/or scores (for example, one modality outputs “sitting upright” posture class, and another one – a skin conductivity value [6]), and these outputs are further combined to produce a “final” emotional category. Fusion can be also done on even lower feature level, that is, sets of features of all modalities are concatenated into one vector, and this vector is fed into a classifier to obtain a “final” category [7]. Third, different reasoning methods can be used for fusion: SVM or decision tree [7], Gaussian process classification [6] and many others.

Forth, some works on emotion recognition proposed to detect transitions between emotional states, for example, to distinguish between onset, apex and offset of emotional states [10]. Thus, it is needed to reason along timeline (on the data at different time moments). Reasoning along timeline can be useful also for synchronizing data of different modalities [7] and for detecting long-lasting emotional states, for example, long-lasting frustration of a student [6].

Last but not least, recently it was proposed to use context in emotion recognition, such as a state of a tutoring dialogue [11] or situation of a person, because same emotions may be expressed differently under pressure to be formal, as in a court, and in relaxed state, as in a party [12]. Also cultural differences in perceiving and expressing emotions exist [13].

Furthermore, studies into multimodal fusion and machine learning in other domains suggested several other ways for improving recognition accuracy, for example, biometrics-based personal authentication was improved by employing user-dependent fusion models in the work [14]. A well-known method to increase accuracy is to employ classifier ensembles [15]. This method aims at overcoming the problem that none of existing machine learning algorithms outperforms the others with all data, and it works as follows: first a large set of algorithms, called classifier pool, is trained on one part of available data and validated on another part. Then, depending on the validation results, an appropriate subset of the trained algorithms is selected at the moment of fusion. Such a subset (called classifier ensemble) can include one or more members and can be selected in different ways.

3. Fusion Framework Implementation

The overview of functionalities, used in different multimodal fusion approaches, is presented in Fig. 1: some researchers employ fusion directly on feature vectors (that is, no optional grey blocks are used), while others employ first or second blocks (e.g., when input components are developed separately). The third option, reasoning along timeline, is used less commonly.

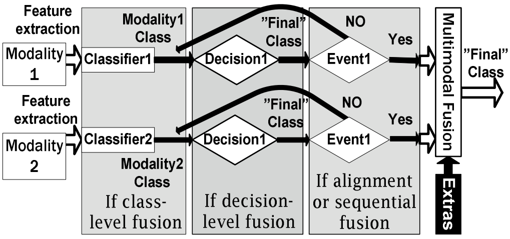


Figure 1: Overview of multimodal fusion for affect recognition. Grey boxes denote optional elements; each of these elements may or may not be included into processing. Grey box “Extras” denotes any additional information used, e.g., results of offline testing for selecting an appropriate classifier ensemble, user’s location, task, nationality, ID etc**.**

In order to provide possibility to flexibly combine functionalities, listed in the previous section and shown in Fig. 1, we implemented a software framework with the architecture presented in Fig. 2. Currently the framework supports only categorical emotional models because they provide intuitive labelling and a freedom to choose any application-specific set of categories and thus are most commonly used [1], but classification can be done along several dimensions simultaneously, thus providing support for PAD models annotated by discrete values. AND/ OR fusion rules, provided by the framework, are useful only for decision-level or class-level fusion for classification with categorical models, but other implemented fusion methods, Weighted Sum and SVM (Support Vector Machines, we use its implementation in TORCH library [16]), can be used at any level of fusion and can be extended to support continuous dimensional models.

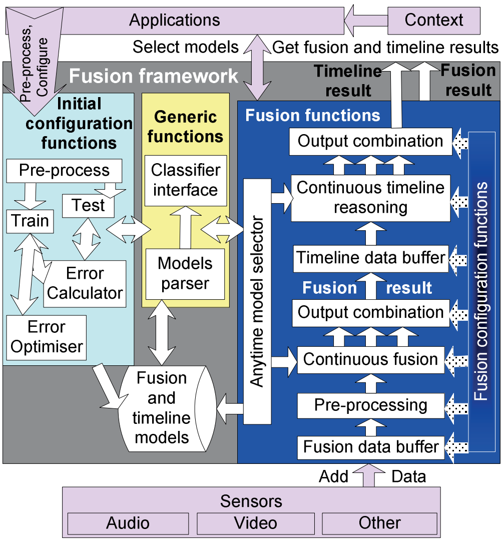


Figure 1: Architecture of the fusion framework.

3.1. Initial configuration and generic blocks

Initial configuration is essentially creation of models which will be used for fusion. All provided fusion methods (AND/ OR rules, Weighed Sum and SVM) can be trained by optimizing errors on training data. Except for SVM, all other fusion methods can be also used without training. Combining several trained and fixed methods together is also possible, for example, combining several SVM or SVM with AND/OR rules.

One of the main generic building blocks of Fusion Framework is Model Parser: it reads templates of trainable models from template files at training stage and ready (non-trainable and already trained) models from model files at fusion and test stages and chains the methods in the specified order depending on the found keywords (such as "IF", "THEN", "SVM", "AND", ">" etc). The actual functionality of the fusion methods and storage of all their parameters (weights and thresholds, for example) are implemented within the second generic building block, called Classifier Interface. Consequently, for specifying the desired fusion functionality it is needed either to create appropriate template files and to train them, or to write models by hand. In some cases pre-processing is required before training and testing, so it is needed to choose from the provided pre-processing functions for combining asynchronous data of different modalities and/ or for data normalization and to place these functions at the start of the methods chain.

For training and testing it is also needed to specify a function for calculating the error cost. This is done within one of the core building blocks of initial configuration, called Error Calculator. Training aims at minimizing the cost function (it can be a weighted sum of misclassifications of each class, for example, where weights represent class importance) while satisfying additional constraints, for example that number of misclassifications of some class should not exceed an application-dependent target value. Although there is no guaranteed way to achieve a desired trade-off between misclassification rates of different classes in multi-class classification problems (especially if number of training samples differs for different classes, which is often the case), using cost functions is a popular approach to this problem [17]. Choice of cost function affects the choice of another core building block of initial configuration, Error Optimizer, an algorithm for minimizing the chosen cost function. Finding SVM model requires solving quadratic optimization problem (implemented in TORCH [16]). Different trade-offs between misclassification rates of different classes can be achieved by varying penalties for misclassification of positive and negative examples. Finding optimal weights and thresholds for other methods is done with differential evolution method [18].

Templates of trainable fusion models include fusion methods, input modalities and output modality, where input modality can be any feature value or a class score:

```
IF speech1 > X
AND audio_volume2 > X
THEN FusionResult=excited
```

In the example above "speech1" denotes a score of recognising class "speech" by method number one (such as by a particular microphone or a particular audio processing algorithm – there may be many microphones or audio processing algorithms employed). Users can choose any string for naming input and output modalities. The keyword "X" denotes that this value should be searched during training; "X" is used always, although actual values will differ from each other after training, when this template will turn into a model:

```
IF speech1 > 0.7
AND audio_volume2 > 0.8
THEN FusionResult=excited
```

Such model can be also written by hand, for example, decision-level models don't require training.

As emotion recognition with categorical models is a multi-class classification task, all models are trained in "one against all" fashion, and templates need to be written for recognizing each emotional state (e.g., "final result" in Fig. 1). Input modalities and fusion methods, used for recognizing different emotional states, may be different, for example, a template for recognizing "user approval" state can look as following:

```
IF SVM(clapping, audio_volume2) > 0
OR SVM(laughing, smiling) > 0
THEN FusionResult=approval
```

Training according to this template results in two sets of support vectors – one set for each SVM.

It is possible to create either a set of models (classifier pool) or just one model for recognizing each state. After models are created, they can be described by a list of descriptors including error rates (useful for selecting a classifier ensemble), user nationality or ID, context in which these models are valid (for example, user task or formal vs. informal situation) and anything else useful for model selection at the moment of fusion. Confusion matrix, test data size and modalities are automatically added to a model description during testing, and other descriptors can be added manually. An example of model description, stored together with each model, is:

```
Modalities:    audio_volume2,    clapping,
laughing, smiling
Test data size: 410
User Nationality: 11
User Situation: concert/theatre
Confusion Matrix:
       approval disapproval neutral total
approval      72         8         20     100
disapproval    5        95         10     110
neutral       10        10        200     220
```

In the example above only three emotional states are listed, but the framework supports any number of output classes, as well as any number of input modalities.

Models for reasoning along timeline are created and described in a same way as fusion models, with just one difference: time intervals (in seconds) between events should be defined. For example, a primitive model for detecting onset of user anger can look as following:

```
IF neutral > 0.8
AND NEXT (0-10) angry > 0.7
THEN TimelineResult=anger_onset
```

Expression “angry > 0.7” denotes that confidence in recognizing “angry” state should exceed 0.7.

Additionally, reasoning along timeline can be done by voting among results within some (usually small) time window. We added this option after the first version of the framework was developed for two-class recognition problems and tested in a simulated task of continuous biometric verification [19]. After these first tests we significantly simplified chaining of different framework functionalities, and now specifying that reasoning along timeline should be done by voting requires only setting of two parameters in the framework configuration file, the interpretation method and the voting time window:

```
Interpretation: voting
InterpretationVotingInterval: 0.1
```

After the first tests the framework was also extended to multi-class multi-dimensional problems in order to provide simultaneous classification of inputs along several dimensions, for example, to estimate at the same time level of pleasure (e.g., neutral, approval or disapproval) and level of excitement (e.g., low, mid or high) – the option valuable for emotion recognition because often it is easier to detect excitement then to estimate level of pleasure, and thus in one-dimensional classification information regarding pleasure may be lost due to relatively low confidence in it. Specifying multi-dimensional classification is also an easy task: the dimension is the model term between the keywords “THEN” and “=” (the examples of fusion models above are all along the only dimension “`FusionResult`”, while two-dimensional classification would be performed if in some models the term “`FusionResult`” would be replaced by the term “`Pleasure`” and in other models – by for example the term “`Arousal`”.

After the first tests we also improved real-time fusion functionalities, described in the next section, provided easier to use options for on-the-fly adaptation of fusion and added pre-processing (such as combination of asynchronous data, normalization etc) functionalities for training, testing and real time fusion stages.

3.2. Fusion

As Fig. 2 shows, fusion framework interface towards individual modalities is very simple: each time when new data is available, it should be put into Fusion Buffer using Add Data function. Data is added in a format “modality name – value – confidence in this value”, which allows to have as many modalities as users want and to use confidence in reasoning. Interface towards applications is also simple: a function for selecting models for fusion and for reasoning along timeline, called Anytime Model Selector, and functions Get Fusion Result and Get Timeline Result that return a corresponding result and a confidence in this result. It is also possible to configure diverse fusion parameters: to select pre-processing options; to specify confidence thresholds and time intervals for keeping the data in each buffer; to chose how to combine outputs of different models (e.g., by voting or by weighted sum).

Fusion is performed continuously (triggered by new data arrival) on the data stored in a Fusion Buffer by models selected by Anytime Model Selector. Results of continuous fusion are stored to Timeline Buffer, and reasoning along timeline is performed, also continuously, on the data in this buffer by models selected by Anytime Model Selector or by voting.

Anytime Model Selector provides a convenient way to adapt to diverse contexts and to select classifier ensembles for improving recognition accuracy or for satisfying specific application requirements. For example, if emotion recognition from speech is running on a user’s mobile phone, adaptation to context can be done by using a model for “informal situation” if a person calls to a spouse, and by switching to another model for “formal situation” at the next moment if the person answers a phone call from his/ her boss.

Anytime Model Selector also provides a convenient way to improve recognition accuracy and/or to adapt to application requirements by selecting a subset of models that have shown the best accuracy for currently available modalities, currently most confident modalities or current values of the modalities: selection of models according to specifics of each data sample (called dynamic classifier selection) is one common way to increase accuracy [15]. If by some reason a certain subset of emotional states is currently more important for an application than the other states, Anytime Model Selector allows to choose the models that have shown the best accuracy in recognizing these particular states. It is also possible to select models with a desired set of modalities if an application has higher trust in them.

When an application calls Anytime Model Selector, it submits a list of descriptors that are compared to model descriptors one by one, and at each step models matching next descriptor are selected from a group of previously selected models. Output Combination block combines outputs of all selected models by either voting or weighted sum of the normalized scores.

4. Experiments

We validated the fusion framework in the tests on audio-video based emotion recognition of an audience in three contexts: in a theatre, circus and a sport event. We tested, how much effort is required to get the desired functionalities (to change a parameter in a framework configuration file, to call some function or to write a piece of code) and to create models, and whether real-time processing of video and audio data, model selection and fusion of asynchronous data work together sufficiently fast. In the experiments we attempted to differentiate between the following situations:

- audience waiting for a start of a show
- audience leaving a show (e.g., during a break)
- moderate approval of a show
- strong approval

Recognition of the last two situations is the main goal if an application is interested in evaluating degree of an interest of the audience, and it is needed to distinguish between these two situations and the first two situations that do not allow to evaluate, whether the audience liked or disliked the show (and thus we consider them as a neutral state of an audience). Naturally an interactive application needs also to recognize an audience' dislike of a show, but we were not able to find such data.

4.1. Data and individual components

For this study we used shots of audience found in movies and TV programs. We found shots of three contexts: in a concert, in a circus and in a basketball match. For each context we were aiming at distinguishing between three emotional states: neutral, moderate approval and strong approval. We also found shots showing how an audience leaves a show, but only in a concert hall. As an audience is almost never shown for a long time, each shot lasted for few seconds. (During a match an audience can be shown for longer time periods, but most of the time a commentator is speaking and thus shots of audience' emotions without the commentator's voice are not long either.) We found 5-10 shots of each emotional state for each context, for examples see Fig. 3. The smallest number of shots was found for "strong approval" of a concert and "leaving a show" situations.

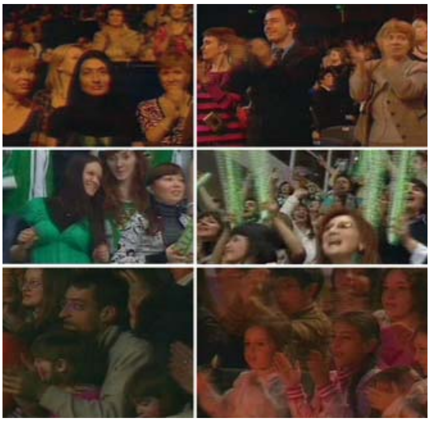


Figure 3: Examples of video shots in the database: "moderate approval" state is presented in the left, "strong approval" – in the right. For each emotional state the top example shows an audience in a concert hall; the middle one – in a sports match; the downmost one– in a circus.

In the tests we projected the shots to a screen and used a web camera facing the screen as video input and a microphone as audio input. Data projection decreases quality of video input, but the problem of emotion recognition from faces in a crowd is not solved by the current state of the art algorithms anyway and thus from video data we used only the optical flow – amount of motion at some moment of time compared to the previous moment (all shots were taken by still cameras).

The video analysis component is based on the widely used OpenCV library [20] and performs several types of processing and analysis on the live video in parallel. The optical flow of the video stream is calculated as the average of motion vectors estimated for each image pixel over a small region around them, representing the average overall motion in the scene at any given time -- and not considering the movement of any one object in particular. The component also provides the face detection functionality, based on the Viola-Jones rapid object detection algorithm [21], so in a future we plan to acquire better quality data and to use a ratio between the number of faces detected and the optical flow as more reliable indicator of the audience' activities.

In our tests optical flow of "strong approval" during a basketball match was much greater than in any other situation and appeared to be a fairly reliable indicator of this situation. Optical flow of "leaving a show" situation was fairly similar to that of "moderate approval", but greater than that of a "waiting of a show start" situation.

Our audio processing component processes live audio and outputs the frame power and the classification result. It classifies the audio stream into eight classes: silence, speech, music, variable and constant noise, whistling (e.g., for recognition of disapproval in a sports match), applauding and clapping (applauding by a few persons only). Audio classification is based on HMM (Hidden Markov Models, we use the implementation in TORCH library [16]) because HMM is a trade-off between accuracy and computational cost and can be well applied to live audio analysis due to its short response time. 29 Mel-frequency cepstral coefficients are calculated using 20 ms time window and fed into HMM models of each class. Class models were trained beforehand on a separately recorded data (mono, 16 bits, 16 kHz sampling frequency) so that number of HMM states, number of Gaussian mixtures and parameters of each class model were optimised for recognising this class.

Power of an audio signal can be used as an indicator of level of excitement of an audience, but it does not allow distinguishing between positive and negative excitement, let alone occasional noises. The power of applauding is a reliable indicator of positive interest, though. Recognition of speech and music classes may allow to postpone emotion recognition until sounds from other sources than audience (such as commentator' speech in a sports match or music in a concert) cease.

Audio component classified "strong approval" and "moderate approval" states in a context of a sport match as "variable noise" (there were no applauding indeed), and "leaving a show" also as "variable noise". "Strong approval" of a circus audience was classified mainly as applauding, but also as "variable noise" and "clapping". "Moderate approval" of a circus audience was classified as "clapping" mainly, but also as "variable noise" and "speech". Audio classification in a concert appeared to be the most difficult task as this audience is the least expressive one: "strong approval" state was classified partly as "clapping" and partly also as "speech", while "moderate approval" – mainly as "speech". "Waiting for a show start" situation was partly classified as "speech" (sometimes viewers talked to each other indeed) and partly as "silence" or "variable noise.

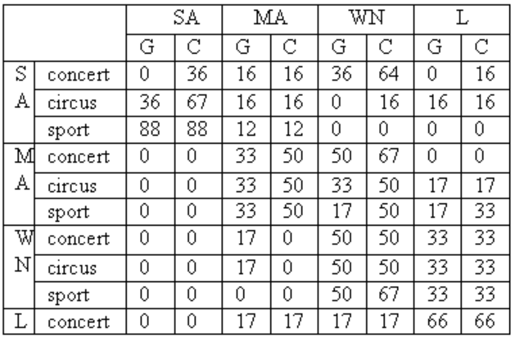
4.2. Fusion

In order to test the functionalities of the fusion framework, we created models using half of the available data and logged results of real-time processing of the other half of the data. We attempted to distinguish between the four above-listed situations with context-independent models (SVM trained on the merged data of all three contexts: concert, circus and match and using a merged feature vector of all audio classes, power of audio signal and strength of optical flow) and with context-dependent models: AND rules created for each of three contexts separately, using own set of modalities for each context. We did not use SVM for context-dependent classification because even amount of merged data for context-independent classification was fairly small for SVM training, and we did not use AND rules for context-independent classification because accuracy of SVM is usually much higher than that of AND rules. Models were first trained and then tested, and after that ensembles of the best models were used in fusion. The capability of the framework to employ classifier ensembles was found to be important in these tests, while attempts to use the only model per situation resulted in very low classification accuracy because none of situations was associated with the only audio class and because of random delays in video processing.

As audio and video components are not fully synchronized, fusion is done each time when new data of any of modalities arrives, using most recent data of all modalities stored in a fusion buffer. (The term "modality" here denotes either one audio class or power of audio signal or optical flow; that is, altogether we had nine modalities). The overall classification of a shot is done by reasoning along timeline and depends on how many times fusion resulted in this class.

Table 1 presents the results of classifying the test shots and shows that "strong approval" in "sport" context was the easiest situation to classify for both generic and context-dependent fusion models. For the majority of other situations context-dependent models have shown higher recognition accuracy despite that AND rule is a fairly primitive fusion method.

Table 1: Confusion matrix presenting percent of correct recognition of four situations in three contexts: "SA" stands for "strong approval"; "MA" – for "moderate approval", "WN" – for "neutral, waiting" and "L" – for "leaving, neutral" states. "G" stands for generic and "C" – for context-dependent models.



		SA		MA		WN		L	
		G	C	G	C	G	C	G	C
SA	concert	0	36	16	16	36	64	0	16
	circus	36	67	16	16	0	16	16	16
	sport	88	88	12	12	0	0	0	0
MA	concert	0	0	33	50	50	67	0	0
	circus	0	0	33	50	33	50	17	17
	sport	0	0	33	50	17	50	17	33
WN	concert	0	0	17	0	50	50	33	33
	circus	0	0	17	0	50	50	33	33
	sport	0	0	0	0	50	67	33	33
L	concert	0	0	17	17	17	17	66	66

5. Conclusion

This paper presented a framework for multimodal real time fusion and the experiments of using this framework for multimodal recognition of emotional states of an audience. Existing works on emotion recognition are not targeted at recognizing emotions of an audience despite that emotions of co-located people affect each other [5] – probably because current video processing algorithms do not recognize facial expressions even of one person in realistic settings [1], not to speak about many faces in a crowd. We found recognition of selected emotions of an audience to be an interesting task which allowed us to test diverse functionalities of the fusion framework. Although we were not particularly aiming at developing a method of recognizing emotional states of an audience, we think that the achieved recognition rates are not too bad considering that the video component provided only optical flow and that audio classification by the audio component was not always in agreement with a human perception.

Furthermore, as video data in our experiments was of low quality and as we used fairly lightweight video and audio analysis methods, as well as simple fusion methods, we think that the presented method can also work on mobile devices, exploiting their context recognition capabilities together with their capability to provide accelerometer data for affect recognition.

However, the main goal of the experiments was to test whether the framework provides all desired functionalities in a fairly convenient way. The experiments confirmed that the framework easily adapts to whatever input data is available; allows to flexibly combine feature-level, class-level and decision-level fusion methods; to reason on data along timeline and to select appropriate classifier ensembles in real time depending on contexts of emotional behaviour and test accuracy. The main goal of the framework development was not to invent any new fusion or classifier ensemble selection methods, but to provide means to experiment with diverse methods in real time settings with no or very little programming effort and this way to facilitate research on affect recognition. The tests have shown that configuring the majority of desired functionalities is easy: it requires either to write a model template or to set a parameter in the framework configuration file.

Although currently adding a new pre-processing functionality requires writing a function, and choosing existing normalisation functionality requires writing several lines of code (currently there are no parameters for it in the framework configuration file), it is a fairly little effort and in a future will be further reduced. Integration of other reasoning methods is also simplified due to the modular framework structure.

The experiments have confirmed that adaptation to context increases recognition accuracy and that employing classifier ensembles for class-level fusion is essential when emotional states can not be associated with the only class. As use of context and dynamics of emotional behaviour are important issues in affect recognition [1], we consider the framework capability to provide these functionalities in a convenient and flexible way as an important advantage. Adaptation to application requirements (for example, choosing a trade-off between recognition errors of different emotional states), although not tested in this work, can be another useful feature. Although it may be needed to store a fairly large set of fusion models in case of adaptation to many different situations, search for fusion models is fairly fast.

Currently the framework capability to reason on dynamics of data is not so advanced, but the architecture of the framework allows integration of other methods of temporal (along timeline) and instant (parallel) fusion. Future work includes integration of Hidden Markov Models for reasoning along timeline, even though HMM can not always outperform simpler methods [7]. Apart from integrating more fusion methods, future work includes research on dealing with errors of input components. Currently fusion framework allows to use confidence in inputs in reasoning, but it does not significantly reduce the number of misclassifications because the confidence in them can be fairly high. How to deal with erroneous inputs is generally a challenging problem of multimodal fusion, and there are not so many solutions proposed.

Despite that the problem of creating appropriate fusion models for each task is the responsibility of application developers (and will remain so in a near future because this is a global problem in machine learning, mainly solved by trial and error approach), the tests presented in this work are encouraging enough to apply the framework also to other fusion tasks.

Acknowledgement: this work was funded by EU Callas project, contract IST – 034800.

References

[1] Zeng, Z., Pantic, M., Roisman, G., Huang, T., A Survey of Affect Recognition Methods: Audio, Visual, and Spontaneous Expressions. IEEE Transactions on Pattern Analysis and Machine Intelligence, 31(1): 39-58, 2009.

[2] Gilroy, S. et al., An emotionally responsive AR art installation. ISMAR 2007.

[3] http://www.cs.waikato.ac.nz/ml/weka/

[4] Lahti, J., et al., Context-aware mobile capture and sharing of video clips, Handbook of Research on Mobile Multimedia, Ibrahim, I. K., Kepler, J. (eds.), 2006

[5] Masthoff, J., Gatt, A., In pursuit of satisfaction and the prevention of embarrassment: affective state in group recommender systems. User Model. User-Adapt. Interact. 16(3-4): 281-319 (2006)

[6] Kapoor, A., Burleson, W., Picard, R., Automatic prediction of frustration, International Journal of Human-Computer Studies, 65(8): 724-736, 2007

[7] Gunes, H., Piccardi, M., Automatic Temporal Segment Detection and Affect Recognition from Face and Body Display, IEEE Transactions on Systems, Man, and Cybernetics – Part B, 39(1): 64-84, 2009.

[8] Clavel, C. et al.. Fear-type emotion recognition for future audio-based surveillance systems. Speech Communication, 50(6): 487-503, 2008.

[9] Caridakis, G., Karpouzis, K., Kollias, S., User and context adaptive neural networks for emotion recognition. Neurocomputing, 71(13-15), 2008.

[10] Pantic, M., Patras, I., Dynamics of Facial Expression: Recognition of Facial Actions and Their Temporal Segments from Face Profile Image Sequences, IEEE Transactions on Systems, Man, and Cybernetics, Part B 36(2): 433-449, 2006.

[11] Forbes-Riley, K., Litman, D., Predicting Emotion in Spoken Dialogue from Multiple Knowledge Sources, HLT/NAACL 2004: 201-208, 2004

[12] Douglas-Cowie, E. et al. The HUMAINE Database: Addressing the Collection and Annotation of Naturalistic and Induced Emotional Data, ACII 2007, LNCS 4738

[13] Kleinsmith, A., De Silva, R., Bianchi-Berthouze, N., Cross-cultural differences in recognizing affect from body posture, Interacting With Computers, 18(6), 2006.

[14] Jain, A., Ross, A., Learning User-Specific Parameters in a multibiometric system, ICIP 2002

[15] Ko, A., Sabourin, R., Britto, A.: From Dynamic Classifier Selection to Dynamic Ensemble Selection, Pat. Recognition 41, 1718-1731, 2008

[16] TORCH, http://www.torch.ch/.

[17] S. Suresh, N. Sundararajan, P. Saratchandra. Risk-sensitive loss functions for sparse multi-category classification problems. Inf. Sciences, 178(12), 2008

[18] Storn, R., Price, K., Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces, Journal of Global Optimization (1997), 11 (4), pp. 341 – 359

[19] Vildjiounaite, E., Kyllönen, V., Software Framework for Multimodal Fusion in Ubiquitous Computing Applications, Network and Service Security Conf. 2009

[20] http://sourceforge.net/projects/opencvlibrary/

[21] Viola, P., Jones, M., Rapid object detection using a boosted cascade of simple features, IEEE Conference on Computer Vision and Pattern Recognition 2001

PUBLICATION VI

Lightweight adaptation of classifiers to users and contexts: trends of the emerging domain

The Scientific World Journal, 2015, Article 434826, 29 p.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2015, Article ID 434826, 29 pages
http://dx.doi.org/10.1155/2015/434826

Review Article

Lightweight Adaptation of Classifiers to Users and Contexts: Trends of the Emerging Domain

Elena Vildjiounaite,[1] Georgy Gimel'farb,[2] Vesa Kyllönen,[1] and Johannes Peltola[1]

[1] *VTT Technical Research Centre of Finland, Kaitoväylä 1, 90571 Oulu, Finland*
[2] *The University of Auckland, Private Bag 92019, Auckland 1149, New Zealand*

Correspondence should be addressed to Elena Vildjiounaite; elena.vildjiounaite@vtt.fi

Received 27 April 2015; Revised 10 July 2015; Accepted 22 July 2015

Academic Editor: Stefanos Kollias

Intelligent computer applications need to adapt their behaviour to contexts and users, but conventional classifier adaptation methods require long data collection and/or training times. Therefore classifier adaptation is often performed as follows: at design time application developers define typical usage contexts and provide reasoning models for each of these contexts, and then at runtime an appropriate model is selected from available ones. Typically, definition of usage contexts and reasoning models heavily relies on domain knowledge. However, in practice many applications are used in so diverse situations that no developer can predict them all and collect for each situation adequate training and test databases. Such applications have to adapt to a new user or unknown context at runtime just from interaction with the user, preferably in fairly *lightweight* ways, that is, requiring limited user effort to collect training data and limited time of performing the adaptation. This paper analyses adaptation trends in several emerging domains and outlines promising ideas, proposed for making multimodal classifiers user-specific and context-specific without significant user efforts, detailed domain knowledge, and/or complete retraining of the classifiers. Based on this analysis, this paper identifies important application characteristics and presents guidelines to consider these characteristics in adaptation design.

1. Introduction

As was observed in [1], conventional data mining is driven by academic interests (e.g., the development of innovative algorithms) rather than by practical considerations. At the same time its applications need to account for the whole process of solving real-world problems, including user interactions and influence of environmental factors. Just the same holds for classification: for example, research into classification methods for pervasive computing largely focused on datasets, collected in research labs or environments, occupied by researchers, and ignored diversity of real-world settings [2]. Systems' evaluation should be also more realistic; for example, not only accuracy should be assessed but also the amount of efforts/resources, spent to achieve this accuracy, should be measured [3].

This paper focuses on practical solutions for runtime situational adaptation of classifiers in cases,where context-independent reasoning either notably decreases the system accuracy or causes considerable users' discontent. Multiple existing surveys of reasoning methods in various research and application areas, for example, [4–7], to cite only a few most recent ones, either do not discuss context adaptation at all or do not distinguish lightweight context adaptation methods from less practical ones. Contrastingly, our review below focuses on techniques, proposed for adaptation to large varieties of users and contexts and requiring neither significant explicit interaction efforts nor detailed domain knowledge. We analyse influence of various context types on classification methods and suggest which adaptation techniques better suit different types of changes in user and system behaviour. The obtained recommendations can be beneficial for designing adaptive systems in frequent practical cases where the successful adaptation could be notably helpful for the users, while the unsuccessful one would not cause serious problems.

The paper is organised as follows. Section 2 discusses limitations of the conventional adaptation approaches; Section 3 presents an overview of the lightweight adaptation; and Section 4 details and compares approaches, proposed in the selected research areas. Recommendations on designing the adaptation and concluding remarks are presented in Sections 5 and 6, respectively.

2. Limitations of Conventional Adaptation Approaches

The majority of classification systems do not provide for diversity of real-life situations. Often, classifiers are developed for the only usage context and heavily rely on knowledge about specifics of this target context, acquired ad hoc from an expert and applied to solutions that can hardly adapt to new conditions [7]. For example, event detection in multimedia analysis systems is often based on recognising small sets of context-specific sounds or visual objects [8–10] and hence reducing dependency of multimedia analysis systems on domain knowledge is a serious challenge [11, 12].

Most common approach to context adaptation is to specify typical situations at design time, to develop a separate classification model for each of these situations, and to define mappings between the situations and corresponding classification models. Then at runtime these mappings are used for model selection. For example, affect recognition systems utilised separate classification models for females and males [13, 14] or for silent and talking users [15], whereas recommender systems employed different reasoning strategies for heterogeneous and homogeneous groups [16, 17]. Similarly, a physical activity recognition system in [18] recognised eight predefined contexts and then selected classification models to recognise sets of context-specific activities: for example, "typing", "sitting," and other activities in a "lab" context were recognised by one set of classifiers, while "sleeping," "eating," "sitting," and other activities in "home" context were recognised by a different classifier set. However, this approach allows only a coarse adaptation and fails in contexts which were not predefined (e.g., real life is not limited to the eight contexts, selected in [18]) and when humans do not behave according to expectations of system developers (as is often the case); for example, emotion expression of a reserve female may be closer to males' ways, and activities may be not so strictly linked to places (e.g., "typing" activity may occur not only in a "lab" context if a person often works at home, in airplanes, etc.).

Accounting for personal differences, social rules, and etiquette is said to be an important goal for recommender systems [19, 20], interactive systems [21, 22], and affect recognisers [23]. Adaptation of information retrieval systems to personal differences has also gained more attention recently [24]. But social rules and personal differences are not strict, are difficult to formulate, and vary significantly depending on person's culture and age. Other examples of elusive contexts are personal goals (e.g., search intents) and variations between environments. Due to the latter, adapting the model to new scenes is listed among the most significant challenges for intelligent environments [2, 25]. As it is virtually impossible for application developers to predefine all usage contexts of their applications [22], adaptation should be performed at runtime using context-specific data.

Recently, involvement of the end users into runtime adaptation process has gained importance [4, 22]. Unfortunately, conventional learning methods do not suit well this purpose. Conventional supervised learning methods require too large datasets for each context to acquire from end user [26], whereas conventional unsupervised and semisupervised methods often employ domain knowledge-based assumptions, which may not hold in all contexts [26]. Furthermore, unsupervised and semisupervised learning schemes favour statistically dominant patterns and thus cannot adapt easily to peculiarities of elusive contexts.

Therefore the runtime context adaptation requires developing methods, capable of learning from small amounts of explicitly and/or implicitly labelled data. If available, unlabelled data shall be also utilised. Recently, such novel techniques were proposed in several application domains: multimedia analysis and retrieval, recommender systems, emotion recognition, and user interaction, but understanding and exploitation of context in fusion systems are still very limited [7]. Below we will describe the concept of lightweight runtime adaptation and summarise suitable approaches.

3. Overview of Lightweight Situational Adaptation Techniques

Situational adaptation to a new context can take two forms: (1) training a model from scratch and (2) modifying a model, which has been already trained on one or more other contexts. In both cases, the adaptation is *lightweight* if its costs are considerably lower than for conventional training of the same application and therefore are acceptable for the end user. The costs include data collection, annotation, and reasoning-related computations. This definition is necessarily informal, because the concept of "user acceptance" hardly can be quantified: it depends on the user's personality, perceived benefits of using the application, convenience of the user-application interaction, and many other nonquantitative psychological factors.

Most of the present systems need large training datasets and intensive computations in order to complete the training process, that is, to estimate all model parameters. Conventional adaptation approaches reduce the need in explicit interaction efforts at the cost of increased need in computational resources: for example, conventional unsupervised and semisupervised learning methods are computationally demanding. The lightweight adaptation of practical interest should rely on a limited user's feedback about the ongoing classification and (possibly) on data of other contexts, to which the system was previously adapted. Therefore the lightweight adaptation should rely on limited modifications rather than total retraining of classifiers; for example, model parameters can be estimated incrementally or partially in order to use the available training data most efficiently. Therefore lightweight adaptation solutions require significantly less explicit interaction efforts than conventional approaches without the need in extra computational resources; most often, lightweight approaches require notably less computational resources than conventional ones.

To the best of our knowledge, research into adaptation has not yet provided guidance on choosing trade-offs between the adaptation costs and achieved accuracies. For example, works studying adaptation of algorithm granularity (such as finer or coarser data clustering) to computational resources and users' needs [27, 28] did not provide guidelines on choosing the adaptation parameters. Works, suggesting that system evaluation should include assessment of efforts, required for achieving system goals [3], did not propose trade-offs either. Furthermore, perception of data labelling difficulty depends on a person [29], and user satisfaction with the algorithm's accuracy depends on many factors, including personal users' attitudes towards adaptation and screen size [30, 31]. For example, even as weak as 50% accurate predictions of user interface (UI) preferences are already beneficial for the users of small devices, whereas for larger screens higher prediction accuracy is required to satisfy the users [30]. Thus we suggest choosing the adaptation granularity as follows: employ first, whenever feasible, the lightweight adaptation and perform the conventional finer one either when the lightweight method fails or during the application idle times if the data necessary for adaptation can be collected with no or only minor (nonannoying) explicit user feedback. But detecting the adaptation failure should rely on the user feedback due to differences in the users' needs and attitudes.

This paper focuses on adaptation of classifiers, combining multimodal data via class- or decision-level fusion, because the adaptation of feature-level fusion is more computationally and data demanding. Inputs to these fusion models, called below "cues," can be context descriptors, audio classes, levels of optical flow, interaction modalities, TV programme metadata, and so forth. These cues are provided by just the same lower-level models in all situations. Usually, it is assumed that the lower-level models were built at design time using a sufficiently large development dataset, collected for a context, somewhat similar to the application usage contexts with respect to the cue types and their statistical properties. Types of the cues in the suggested adaptation approaches should be predefined at design time, but exact sets of the cues should not necessarily be predefined: it depends on the chosen reasoning methods. Such a multimodal fusion adaptation may fail if the context change causes poor performance of the low-level models. On the other hand, when adaptation cannot be performed on all levels (e.g., due to insufficient amount of training data), updating top levels is more efficient than updating the lower ones [32]. Also, it is easier for the users to provide explicit feedback on the final classification result, and this feedback is more reliable than the feedback on outputs of lower-level models because these outputs may be unclear to the users or perceived as irrelevant. Employing the feedback on the classification results for updating the lower-level models is not an easy matter as well, due to its nonstraightforward propagation to the lower levels: for example, different modalities may not contribute to the final result in the same way in different contexts.

3.1. Context Types. The terms "situation" and "context" are often used interchangeably; these terms refer to some kind of external or latent factors influencing users' or system's behaviour. In particular, these terms may specify not only fine descriptors, such as time and noise, but also higher-level abstractions like events, locations, names of databases, and so forth. Most often the classifiers are adapted to the context types representing historical, social, task, environmental, and computational factors.

(i) Historical factors embrace anything in the past that may affect current state, for example, user or system actions, changes in user's mood or appearance over time, and recently viewed movies.

(ii) Social factors include rules and customs of interaction between humans, for example, gender/age-dependent behaviour and what is considered polite in different situations.

(iii) Task factors present specific users' objectives, for example, purpose of information search and available time.

(iv) Environmental factors are anything in surroundings that may affect sensor readings, for example, background noise and light.

(v) Computational factors specify system settings, for example, availability or quality of a certain data type (such as image resolution), computational power, and algorithm capabilities.

The most popular context model, called a "representational view" [33], describes the context by a set of features, defined at design time. An alternative more difficult and less common "interactional view" [33] assumes that the context cannot be described with a predefined feature set. Instead, the scope of descriptors is defined dynamically during human activities.

The lightweight runtime adaptation to previously unseen contexts typically employs a mixed model in Figure 1: the fine descriptors ("context cues" in Figure 1) or their types are predefined at design time, whereas the higher-level ones (the "situations" in Figure 1) are defined dynamically at runtime. Dynamic definition of high-level contexts can be achieved via analysis of primary data, for example, segmentation [34] or matrix factorisation [35]. The context change can be also detected via analysis of external factors, for example, context features or user interaction (e.g., users may explicitly declare the change by naming a new context or implicitly indicate it by correcting classification errors and requesting adaptation). Below we assume that analysis of external factors was employed.

Most often, the contextual factors (especially, the high-level ones) are nondiscriminative with respect to a classification task at hand; that is, they do not directly help to classify data; rather, situational changes cause certain changes in primary data. In some cases context may influence the user behaviour: for example, humans usually are freer in

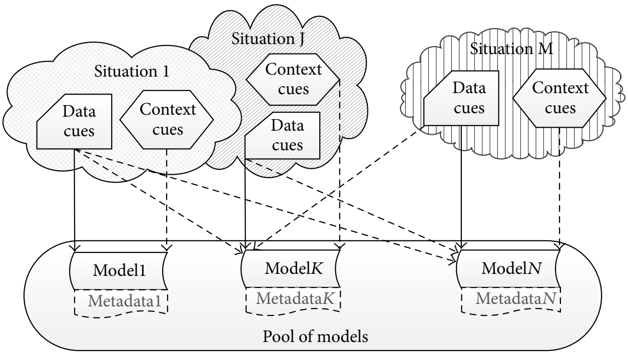


Figure 1: Modelling situation-dependency scenario; dashed lines denote optional data.

expressing feelings when conversing with close friends than with officials. In other cases context may influence internal system functionality: for example, success of audio/visual analysis depends on background. Changes in the primary data cues can be categorised as follows:

(i) *Meaning Changes*. Just the same input cues have to be interpreted differently in different contexts. Often, meaning of cues depends on social factors: for example, whistling that indicates game highlight in basketball matches is a meaningless sound in tennis [8]. Meaning of the cues may depend on historical factors, too (e.g., the users' laughter during a dialog may be interpreted as either happy or sarcastic, depending on the previous statements).

(ii) *Influence Changes*. Importance of the same input cues may vary in different contexts (often, also due to social factors): for example, presence of young children may strongly affect the choice of TV programmes to watch in one family, whereas adults may dominate in another family. Task factors play this role, too: for example, noise or illumination cues may be more or less important, depending on a search goal.

(iii) *Accuracy Changes*. The same input cues may be recognised more or less reliably in different contexts, often due to computational factors; for example, the accuracy of image analysis depends on image resolution. When sensor data are collected in uncontrolled conditions, the environmental factors significantly influence the accuracy of input cues: for example, image background affects the accuracy of object detection. Historical factors may degrade the accuracies, too, when the models become outdated; for example, growing hair may decrease the face recognition accuracy.

(iv) *Availability Changes*. The same input cues may be abundant in some situations and missing in others. Most often this happens in uncontrolled conditions: for example, results of video analysis may be unavailable if users bypass a camera. Social factors may cause incomplete data, too: if it is polite to stay silent, audio cues will be unavailable.

3.2. Typical Adaptation Goals, Data Acquisition, and Reasoning Methods. **Multimedia analysis and retrieval systems** aim at detecting various events or concepts, usually, by trained classifiers. An overview of conventional reasoning methods in this area can be found in the earlier surveys [9, 11, 24, 36]. The detection has to be adapted to both task and computational contexts, that is, to different user queries and different multimedia databases, respectively. Differences between the multimedia databases present a challenge because the algorithms, trained on one multimedia type (e.g., genre) or source (TV channel) and tested on another type/source, are usually 1.5–2 times less accurate than the corresponding within-type/source ones [37, 38]. The inter-user differences present an additional challenge: the adaptation to user queries has to be very quick. Therefore, the retrieval systems are often hierarchical: their lower layers perform context- and user-independent multimedia analysis, and their upper ones are iteratively adapted by reranking results of the lower layers [36]. Event and concept detection also have to be adapted to environmental context because they significantly depend on the background.

All these contexts are difficult to predefine. Therefore in this research area several methods to use fairly small datasets either for runtime training of models from scratch or for knowledge transfer between contexts were proposed. Often, explicit data is collected by asking the users to select which items to annotate and/or to correct system outputs. Hence obtained explicit data may be noisy, but an alternative approach, to provide labels on system-selected items from scratch, is more tiresome and may give more than 10% of errors [39]. Investigation of implicit information gathering approaches gained more attention recently [24], but implicit feedback is less reliable than explicit one because clicked results are not always relevant to the user's search [24]. Noise in datasets, however, is most often dealt with in a fairly straightforward way: by employing generic noise-tolerant classifiers, such as support vector machines (SVMs) [36, 40].

Recommender systems aim at finding items, most interesting for the target users in the target context. An overview of conventional reasoning methods in this area can be found in the earlier surveys [19, 41, 42]. Mostly, the recommenders adapt to the differences between users' personalities, to influence of context on interests of individuals, and to social context, but usually these contexts have to be predefined: for example, the adaptation to the user's goal (task context) is often performed by predefining a few typical contexts (e.g., watching a movie at home versus in cinema). Many recommender systems employ collaborative filtering (CF), a lazy reasoning method, based on the assumption that users, similar in the past, will also remain similar to some extent in future. Accordingly, the CF searches for users, similar to the target user, and creates recommendations for the target user by combining choices of similar users. Frequently, the CF adapts to contexts by so-called "pre-filtering" [43]: searching for similar users only in the data of target context. Hence recommendations can only be provided for users, who already expressed their preferences for predefined contexts, matching a target context either exactly or in a generalised form [43]. Therefore developing better understanding of how to use the context in the recommender systems is an important but largely unsolved problem [43].

The adaptation to the social context is considered more challenging than to the individuals, especially if the groups include people with significantly different personal preferences [44, 45]. Many researchers aim at utilising in the social context knowledge, acquired for a "being alone" context [20, 44, 46–49], whereas others develop methods to provide for heterogeneity between group members by predefining its typical degrees [16, 17] or via negotiations [50]. Also, it was suggested to exploit user preferences, acquired for one domain or task, for enhancing adaptation to a new task: for example, to use preferences for books in a movie recommender [51]. Therefore in this research area methods to transfer knowledge between contexts were proposed. As a group is more than the sum of its members [48], methods for using identities of group members as features [52] and methods to explore personality traits as features [53] were also explored in this area.

The recommenders can acquire training data explicitly, by asking the users to rank items, for example, movies, or their attributes, for example, genres and actors. The context, for which these preferences are acquired, is also often obtained explicitly or via sensors. Some systems instead acquire implicit user feedback by observing how the users deal with the recommended items: for example, select or skip them and view/listen fully or partially [54–56].

Affective computing aims at recognising human emotions, usually, by trained classifiers. Recognition results can be employed in *human-computer interaction, multimedia analysis*, and so forth. Due to the difficulty to understand human behaviour, data in this domain are usually labelled explicitly.

The affect recognition has to be adapted to personal differences and social rules because relations between communicating persons and customary ways to behave in different settings vary significantly. For example, in one context upset persons may scream and grimace, whereas in another context they may stay silent due to etiquette, the emotions having to be recognised only from facial expression [57]. Due to the difficulties to collect and annotate contextual data, however, the majority of human affect analysers remain context insensitive, as the recent surveys state [23, 58]. Most often, they are trained with data, acquired in very limited sets of contexts, and do not generalise well to other contexts. For example, capabilities of an audio-based emotion classifier to recognise several emotional categories (e.g., joy and anger) and distinguish between positive and negative arousal and valence were compared in [59] on six databases of spontaneous, induced, or acted emotions, collected in different countries. This evaluation has shown that the "*performance is decreased dramatically when operating cross-corpora-wise,*" mainly due to differences in displaying spontaneous and acted emotions in different contexts. Therefore this area offers methods of runtime training of context-specific models and methods of knowledge transfer between contexts. Adaptation to historical factors is also studied fairly often, mainly, to previous emotional states, being strongly interdependent with the current one (e.g., an excited person does not calm down instantly).

User interaction (UI) is concerned with providing convenient application interfaces. According to a recent survey [60], adaptation to personal preferences is an important future research direction. Adaptation to various definable and indefinable task factors (such as standing versus walking and answering a call versus a text message); social factors (e.g., in public speech interaction may be undesirable); and environmental factors (e.g., light) is also important [61]. Currently, the UI is mainly adapted to predefined computational contexts, such as screen size and device capability to deliver information via certain modality, for example, audio or video. Adaptation is most often based on rules, created by the application designers or end users: the former ignore the personal differences and the latter require the user efforts. Hence a recent review [60] suggested that interaction adaptation requires fundamental improvements, based on machine learning techniques. Nevertheless this area offers a few studies into knowledge transfer between contexts. Training data is acquired either implicitly via tracking customisation choices or explicitly by asking users to rank options or to perform certain tasks.

4. Basic Approaches and Examples of the Lightweight Adaptation

The lightweight adaptation is an emerging research area with a handful of approaches proposed to date. The lightweight adaptation is most beneficial for the application domains where (1) variety of users and contexts is large and (2) reasoning errors cause no serious problems for the users. This is usually the case with multimedia retrieval and recommender systems. Some additional insights into this problem can be found also in studies into multimodal fusion in other domains, for example, biometrics, and will be presented in this review, too. Although the context influence on human and algorithm behaviour is of the main concern of this paper, the adaptation to differences in users' personalities will be presented, too. For example, affect recognition should take into account that humans usually express emotions freer in informal than in formal settings. However, a reserved person always expresses emotions more subtly than an open one. In this case the adaptations to personal differences and "formal versus informal" context have no conceptual differences.

Some common approaches to decrease the user efforts, such as popular unsupervised and semisupervised learning methods, will not be surveyed here because they either require excessive computations or use too inflexible modelling assumptions. For example, semisupervised learning is often based on the assumption that points, located one near another in the feature space, belong to the same class [26]. However, it is well known that points, which are close to each other in one context, may appear quite distant in another context. This is why modification of similarity measure is a fairly common way to adapt to users and contexts [62, 63]. Furthermore, experimental comparison of conventional semisupervised learning with more lightweight context adaptation demonstrated that the latter can be significantly more accurate [37]. Unsupervised learning favours typical (statistically significant) data patterns and thus may fail to catch atypical context-dependent behaviours. Active learning methods are not surveyed, too, because they choose data samples, most informative for classification, but not necessarily easy for humans to annotate. Moreover, the perception of labelling difficulty depends on a person [29] and thus for end users it is more convenient to choose themselves which samples to annotate.

4.1. Classification of Adaptation Approaches. Context adaptation can be considered a generic machine learning problem of designing and training systems to perform well enough on data, acquired at runtime. However, while conventional learning assumes the similar enough runtime and training data, the situational adaptation may also require accounting for significant differences between the contexts. In designing a conventional system, one decides first whether to use a single classifier or an ensemble of multiple classifiers [64]. In the latter case, additional decisions are to be taken at the four design levels:

(i) Combination level: how to deal with outputs of the base classifiers (members of the ensemble), in particular how to select these members for each case and/or how to combine their outputs.

(ii) Classifier level: which base classifiers to choose, for example, whether to employ same or different algorithms.

(iii) Feature level: whether the same or different input features should be utilised by all the base classifiers and which features should be chosen.

(iv) Data level: whether the same or different datasets should be used to train all the base classifiers and, if needed, how to choose the training data to select the best classifiers and/or optimise their combination.

Context-adaptive systems can use a single classifier or multiple classifiers, too, but additional decisions on using contextual parameters are needed in both cases. The context data in a single classifier can serve as an input feature or a latent variable. A multiple classifier system offers more choices: for example, either a single context-specific model can be trained for each context or multiple models can be trained for each context. Some or all the classifiers can also use the context as a feature or latent variable. These choices are very important because they determine the adaptation type: a multiple classifier system, training its own model for each context, can switch arbitrarily and abruptly between the contexts, whereas a single classifier system, using context parameters as features, can react more smoothly at context changes. Due to the need to reduce user efforts, it is also important to decide whether the data or models of initial contexts can be reused or only the target context data should be used for adaptation to this context: reusing data or models of initial contexts decreases the need in the target context data but may hinder the adaptation if the initial and target contexts differ significantly.

One of the proposed ways to classify adaptation approaches is to consider a number of employed inference models: adaptation in a multimodel system is a procedure of switching between models, whereas adaptation in a monomodel system is a procedure of tuning its parameters [65]. Another way [7] is to consider use of contextual data: context can be used as constraints (forbidden operations, probabilistic conditioning, etc.) or as additional features, semantics, or situation elements (e.g., context may change a meaning of information or bring new dimensionality into a problem). The work [65] does not list adaptation via selecting multiple models and combining their results, though, and the work [7] is concerned with use of context features rather than high-level situations. In addition, when a totally new situation emerges, it may be needed to train a new model from scratch.

We suggest classifying situational adaptation approaches based on the choices on combination, data, and feature level, and we suggest three major groups, called in brief *model selection*, *ensembles*, and *context as a feature*. *Model selection* group encompasses procedures of using contextual data for selecting a single model from existing models, as well as methods to train a new model from scratch. *Ensembles* group encompasses procedures of selecting one or several models from existing models based on their accuracy rather than contextual data, as well as methods to combine their results. *Context as a feature* group encompasses methods to use contextual data as input features (additional dimensions) or latent/hidden variables. Figure 2 presents most interesting approaches from these groups along with our understanding of their ability to handle situations, emerging at runtime. The following sections give more details. Other choices on the classifier and feature levels are influenced by the aforementioned choices and by peculiarities of a problem at hand in ways similar to the conventional systems. The majority of the lightweight adaptation methods presume the same feature sets in all the contexts, but some approaches allow for employing the same feature extraction methods in all the contexts even if this results in context-specific sets of cues.

4.2. Model Selection. The *model selection* group embraces multiple classifier systems, where only one model is trained

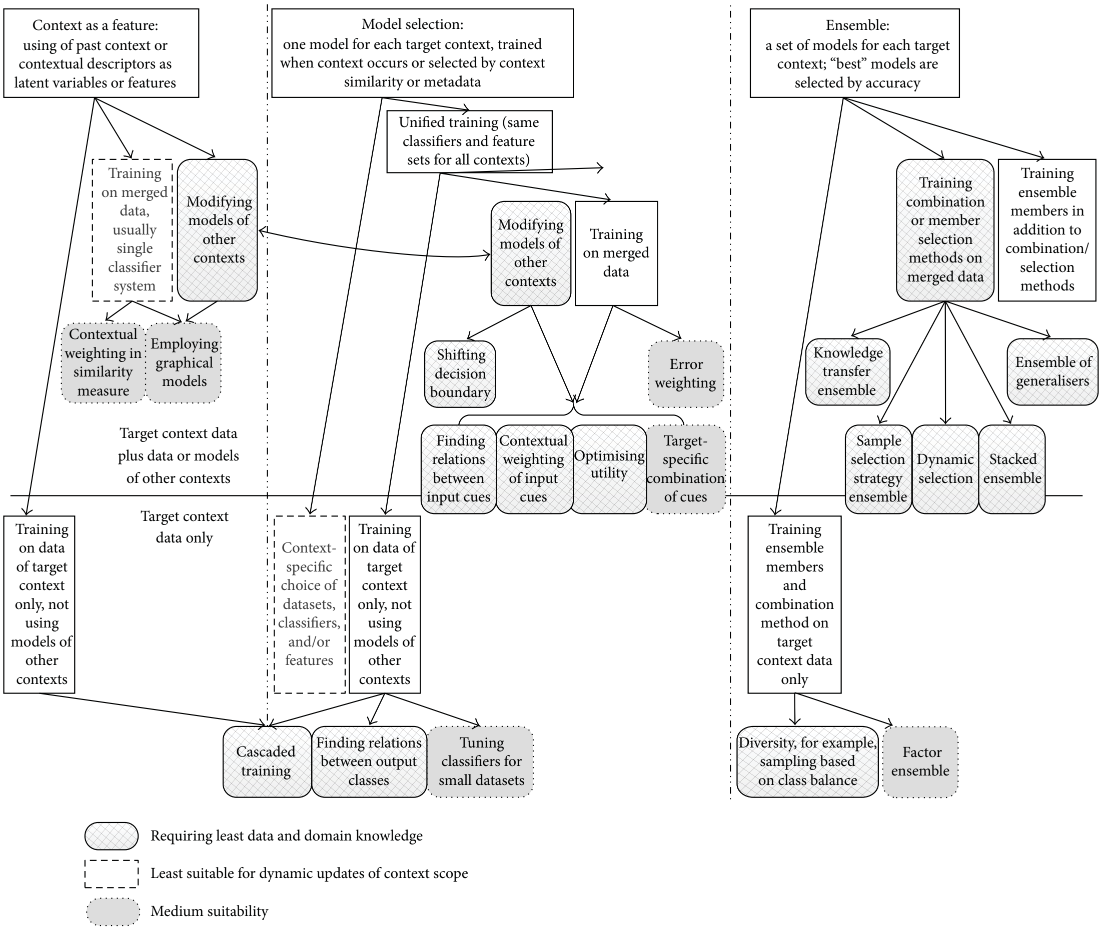

Context as a feature:
using of past context or contextual descriptors as latent variables or features
Model selection:
one model for each target context, trained when context occurs or selected by context similarity or metadata
Ensemble:
a set of models for each target context; "best" models are selected by accuracy
Unified training (same classifiers and feature sets for all contexts)
Training on merged data, usually single classifier system
Modifying models of other contexts
Modifying models of other contexts
Training on merged data
Training combination or member selection methods on merged data
Training ensemble members in addition to combination/ selection methods
Contextual weighting in similarity measure
Employing graphical models
Shifting decision boundary
Error weighting
Knowledge transfer ensemble
Ensemble of generalisers
Target context data plus data or models of other contexts
Finding relations between input cues
Contextual weighting of input cues
Optimising utility
Target-specific combination of cues
Sample selection strategy ensemble
Dynamic selection
Stacked ensemble
Target context data only
Training on data of target context only, not using models of other contexts
Context-specific choice of datasets, classifiers, and/or features
Training on data of target context only, not using models of other contexts
Training ensemble members and combination method on target context data only
Cascaded training
Finding relations between output classes
Tuning classifiers for small datasets
Diversity, for example, sampling based on class balance
Factor ensemble
Requiring least data and domain knowledge
Least suitable for dynamic updates of context scope
Medium suitability

Figure 2: Lightweight adaptation approaches.

for each context. Metadata or contextual values, describing the context each model was trained for, can be stored along with the models to retrieve them by metadata or context similarity. The contexts can be recognised from sensor data or user interaction (the latter gives only a coarse situational description). If the context cannot be recognised or is unlikely to emerge again, the old models can be discarded. The models can also be built for overlapping or nested contexts to either retrieve the best match or combine several suitable models. Depending on the data usage, this group can be further categorised as follows:

(i) *Context-Specific Classifiers and Data*. Each model is trained on the target context data and context-specific features and/or reasoning methods are employed. Lazy methods search for similar cases only within the target context, using context-specific distance measures.

(ii) *Context-Specific Models and Data*. Each model is trained on the target context data, but feature selection and reasoning methods are the same for all contexts. Lazy methods search for similar cases only within an appropriate data part, but the distance measure is the same for all contexts. This category may be less adaptive than the previous one, but it is more lightweight because although the models have to be retrained from scratch for each new situation, no additional designer efforts and data are required for the feature and/or algorithm selection.

(iii) *Context-Specific Models with Mixed Data, That Is, Knowledge Transfer*. Feature selection and reasoning methods are same for all contexts, and knowledge of other contexts is used to build models for the target context; for example, models can be trained on the merged data. More sophisticated methods of knowledge transfer on data, feature, combination, and model levels were also proposed, mainly in research areas called "transfer learning" and "domain adaptation" (in these areas the terms "domain" and "task" are typically used instead of "context"). Data- and feature-level knowledge transfer methods are usually computationally expensive (see, e.g., [5, 37, 66]). Knowledge transfer on combination level is usually done by fusing together outputs of the models, trained for different contexts, for example, with weights depending on context similarities. To obtain a target context model by the model-level knowledge transfer, selected parameters of models for initial contexts are modified. Lazy methods search for similar cases within the data from different contexts and treat samples from these contexts in a different way. Knowledge transfer saves data collection efforts significantly, and the model-level one further reduces data collection needs and training time because usually not all the parameters of the initial models are modified.

While the first two of these approaches suit both similar and dissimilar contexts, the knowledge transfer suitability to dissimilar contexts depends on the transfer type and degree of preserving the old knowledge, for example, on constraints on modified model parameters.

4.2.1. Context-Specific Models and Data. (1) *Contextual weighting* employs n generic cues, or individual data modalities, with outputs S_i, and combines them linearly, $\sum_{i=1}^{n} w_i S_i$, with context-specific weights w_i. This approach is often used in audio-visual data analysis [11, 25] and fusion of data from multiple sensors [7]. For predefined situations the weights can be determined from the prior knowledge; for example, the video modality can be assigned higher weight than the audio modality for daytime and lower at night. The weights can be also calculated using quantitative estimates of the accuracy a_i of each modality in the target context: $w_i = a_i / \sum_{j=1}^{n} a_j$.

Alternatively, for adapting to changes in reliabilities of modalities without relying on domain knowledge and training data, the weights can be estimated from stream entropy [25] or relations between outputs of the different modalities, for example, the variances of several topmost scores [25], using rules of the kind "if the topmost and the second best scores differ less than some threshold, then the weight w_i is high... else...." A fairly small training dataset can suffice for either the accuracy estimation or the rules' derivation.

The lightweight adaptation to a user query in the multimedia retrieval tasks is performed via either "feature relevance estimation," that is, modifying the feature weights, w_i, in a linear similarity measure, or "query vector modification," that is, adjusting the feature weights in a query vector [36]. The weights can be adapted heuristically or by genetic algorithms; the latter require training data, but no domain knowledge. Cues can be combined also in other ways than a simple weighted sum. For example, Calumby et al. [67] employed genetic search for the best combination of various text-, colour-, and texture-based features and their functions including, in particular, products and square roots. This search aimed at minimising the classification error for the labelled items, and the training data consisted of 55 images. A few dozens of user-labelled items are fairly typical database size in multimedia retrieval as users do not provide abundant training data. Therefore contextual weighting is a fairly common approach because, for example, probabilistic approaches, such as Bayesian inference, require more feedback data [68].

(2) *Optimising utility function* is achieved by selecting from a large set of generic cues a subset with the highest utility. Utilities of each subset depend on the target context and can be easily computed. Usually, the utility functions are application-specific sums of context-dependent (often, heuristic) values S_i, reflecting gains or losses of including different cues and/or their combinations in the subset. Adaptation is based on calculating the utilities of various combinations $\sum_{i\in C} S_i$ and optimising (possibly with constraints) the result to choose the subset C_{opt}. This approach does not require training data. For example, for GUI (graphical user interface) adaptation task values S_i may reflect costs of satisfying/ignoring user preferences, easiness of navigating between GUI elements, and so forth, and the optimisation may result in selecting, for example, a picture box and a small text box with a scroll bar in one context and selecting a large text box with no scroll bar in another context.

This approach was employed for user interface adaptation [69–71]. In [69] GUI was adapted to different screen sizes and to user preferences and abilities, for example, special needs of motor-impaired users. Costs and constraints of choosing different elements were estimated by tracing performance of each user, for example, speed of clicking on interface elements of different sizes. However, the constraints elicitation required fairly long and diverse interaction histories: in the tests able-bodied and motor-impaired participants had to perform tasks during at least 25 and 30–90 minutes, respectively [72]. The interfaces were adapted only to fairly similar task contexts, like controlling light intensity, ventilator, and audio-visual equipment in a classroom, and only for an individual GUI usage.

A greater variety of contexts and interaction modalities were considered in [70]: contexts included environmental, task, and computational factors, such as light, weather, noise, motion, screen size, and keyboard type, and I/O modalities included eye tracking, gestures, audio, video, and vibration. Explicit user preferences were acquired by asking the users to manually assign numerical scores for different interface elements in various contexts. This process is not very easy for the users, however, and more error-prone than, for example, selecting the most appropriate elements from available options. Hence in the work in interface adaptation to different platforms [71], utilities of different GUI options (e.g., different font sizes) were partially obtained from users and partially specified by system designers.

(3) *Tuning classifiers for small datasets*: special efforts are taken for selecting data features, classifier parameters, or training samples to reduce negative effect of small data size. This approach was proposed for multimedia retrieval with

SVM (support vector machine), and tuning was performed by selecting SVM kernel or subset of training items [63, 73]. SVM training was done for each query in a standard way, by minimising the classification error for the user-labelled items. In particular, an iterative user's feedback in [73] required every user to label nine retrieved images per iteration and achieved a reasonable precision after 6–8 iterations. Video retrieval in [63] gave the users 15 minutes per query for evaluating and labelling the obtained query results.

(4) *Cascaded training* uses first unlabelled data for initial parameter estimates and then the labelled data for fine-tuning. This approach was applied to deep neural network [74], discrete HMM (hidden Markov model) [75], and MLP (multilayer perceptron) [26] based classifiers. The work [74] mainly aimed at increasing accuracy of offline training; difficulties to obtain the labelled data were not of main concern. The work [26] reviewed approaches for reducing the need in labelled data, but also mainly for offline training. The work [75] was concerned with user-controlled runtime adaptation to indefinable situations (social behaviours) and therefore aimed at finding a quick and lightweight adaptation method. The main goal was to detect show highlights by recognising arousal of a show audience. Classification was performed by HMM, employing the cues from audio-visual analysis (such as laughter, speech, silence, noise, and human motion) as observations. The proposed classifier needed to distinguish between three arousal levels in fairly dissimilar contexts: show types with notably different ways to express excitement (a concert, a circus, and a sport match). Furthermore, the data were collected in uncontrolled settings, and the classifier had to deal with considerably different audio-visual backgrounds and missing behavioural cues. Hence a new model was trained each time after a new context emerged and the user provided annotated examples.

During the first stage an HMM model was obtained for each context in conventional way, by using Baum-Welch algorithm. During the second stage of cascaded training only the observational probabilities of the HMM were optimised with a differential evolutionary algorithm, aiming at minimising the classification error for the labelled data (see also Section 4.2.2). The HMM employed the maximum posterior marginal (MPM) decisions, rather than the conventional maximum *a posteriori* (MAP) ones, to achieve more robust adaptation. As a result, the adaptation did not require significant time and efforts: the annotations were collected for 10 minutes per context, and the annotator was free in choosing samples to label. In the tests as little as 25 labelled samples per context (5–12 samples per class in each context) allowed significantly increasing the classification accuracy, comparing with the conventional HMM training. Comparing with an alternative full-scale adaptation, this fusion-level-only one relieves the users of having to make considerable efforts for recognising and labelling selected behavioural cues in the data, containing large number of mixed sounds and video backgrounds.

(5) *Learning context-specific relations between the outputs of a multiclass classifier* [76] is proposed for multimedia analysis. Because the concept detection is a multiclass classification with nonexclusive classes, its accuracy can be increased by learning the most frequent cooccurrences of classes in different contexts. In the hierarchical system in [76] context-independent concept (class) detection models are trained first. Then an affinity graph with a node for each concept is built to learn context-dependent relations between the concepts. Each two correlated nodes form an edge, the six strongest edges for each node being kept. How many training samples are needed to learn such a graph was not mentioned in [76], but as a whole, the context-specific learning was very fast: less than a minute comparing to tens or hundreds of hours, required usually to completely retrain the concept detection models.

4.2.2. Model-Level Knowledge Transfer.

A lightweight model-level knowledge transfer for trained classifiers is most often done by shifting a decision boundary. Transfer design requires making three major choices: (1) which parameters of initial models are modified (all or just certain selected parameters); (2) how parameters are modified (choice of an objective function and an algorithm to optimise it); and (3) which initial models are modified (e.g., a model trained on the merged data of all previous contexts or a model of a certain context).

(1) Optimising model parameters: two fairly generic ways to shift decision boundary of trained models have been proposed to date: evolutionary algorithms and gradient descent based search for changes in model parameters, minimising classification error in target context. Most often, in this approach only training data for the target context is employed. To increase accuracy of interactive image segmentation, evolutionary optimisation of segmentation parameters in [77] was based on the user's feedback on whether too many or few image segments were obtained. At each iteration the users were presented with a small number of images, for example, six images, and on average 7–10 steps allowed to achieve satisfactory segmentation accuracy.

The evolutionary optimisation was used also in [75] for modifying a discrete HMM based on the MPM decisions. This work studied two approaches to adapt an affect recognition system: cascaded training (see Section 4.2.1) and model-level knowledge transfer. In cascaded training initial models for each target context were trained in conventional unsupervised way by Baum-Welch algorithm. In model-level knowledge transfer a model, trained on data of some other context, served as initial model for the target context (the contexts are described in Section 4.2.1). In both approaches the initial models were modified by evolutionary algorithm to increase accuracy of recognising arousal levels. Let the goal arousal levels be associated with K hidden states $\{\theta_k: k = 1, \ldots, K\}$ and let $\mathbf{X}$ denote a space of the observed vectors $\mathbf{x}$ of cues for each state θ_k. Given observational probabilities $\mathbf{p}_{\text{obs}} = [p(\mathbf{x} \mid \theta_k; \boldsymbol{\alpha}_k): k = 1, \ldots, K;\ \mathbf{x} \in \mathbf{X}]$ with the unknown parameters $\mathbf{A} = \{\boldsymbol{\alpha}_k: k = 1, \ldots, K\}$, the initial training was used to estimate the K^2 interstate transitional probabilities $\mathbf{p}_{\text{tr}} = [p(\theta_k \mid \theta_l): k, l = 1, \ldots, K]$ and parameters, $\mathbf{A}$, of the observational probabilities for the HMM. Then the evolutionary algorithm modified only the observational probabilities to minimise the number of classification errors on the user-labelled data for the target context.

In the tests as little as 25 labelled samples per each target context considerably increased the classification accuracy in comparison with the nonadaptive or context-independent models, despite a fairly significant difference between initial and target contexts.

Unlike the work [75], Caridakis et al. [78] studied emotion recognition of individuals, and in their data contexts did not significantly differ from each other: all the data consisted of records of persons, communicating with four artificial computer characters [79]. The emotional expressions were not very intense because all the records were acquired in the same laboratory and communications with the artificial characters do not follow exactly the same social rules as with real humans [79]. The records were classified into K classes of emotions by a neural network (NN). Its parameters, $\mathbf{w}$, had been learnt initially in a fully supervised mode using a special training set of the labelled records. For each input vector $\mathbf{x}$ of cues, the NN forms the K-component output vector $\mathbf{f}_{\mathbf{w}}(\mathbf{x}) = [p(k \mid \mathbf{x})\colon k = 1, \ldots, K;\ \sum_{k=1}^{K} p(k \mid \mathbf{x}) = 1]$ of class probabilities. To adapt to a new target context, the initial parameters are changed incrementally by minimising, with the gradient descent search, a weighted total error $\varepsilon = \varepsilon_{\text{target}} + \lambda\varepsilon_{\text{ini}}$ on the initial $(\mathbf{x}_{\text{ini}:t})$ and target $(\mathbf{x}_t)$ labelled inputs, the weight λ determining significance of the target context data for adapting the NN:

$$\begin{aligned} \varepsilon_{\text{target}} &= \frac{1}{2} \sum_{t=1}^{N_{\text{target}}} \left| \mathbf{f}_{\mathbf{w}}\left(\mathbf{x}_t\right) - \mathbf{p}_{\text{target}:t} \right|, \\ \varepsilon_{\text{ini}} &= \frac{1}{2} \sum_{t=1}^{N_{\text{ini}}} \left| \mathbf{f}_{\mathbf{w}}\left(\mathbf{x}_{\text{ini}:t}\right) - \mathbf{p}_{\text{ini}:t} \right|, \end{aligned} \tag{1}$$

where $\mathbf{p}_{\text{target}:t}$ and $\mathbf{p}_{\text{ini}:t}$ are the desired output K-component probability vectors for the labelled inputs, $|\cdots|$ denotes the vector norm $|\mathbf{z}| = \sqrt{z_1^2 + \cdots + z_K^2}$, and N_{ini} and N_{target} are cardinalities of the available initial and current training datasets:

$$D_{\text{ini}} = \left\{ \left(\mathbf{x}_{\text{ini}:t}, \mathbf{p}_{\text{ini}:t} \right) : t = 1, \ldots, N_{\text{ini}} \right\}, \tag{2}$$

$$D_{\text{target}} = \left\{ \left(\mathbf{x}_t, \mathbf{p}_t \right) : t = 1, \ldots, N_{\text{target}} \right\}, \tag{3}$$

respectively.

The adaptation was lightweight because only small increments of the weights $\mathbf{w}$ were allowed, and nonlinear signal transformations in each neuron were linearized using first-order Taylor's series decomposition. Due to linearization, the increments of the NN parameters were obtained by solving a system of linear equations with coefficients depending on the initial parameters and all the training data in the datasets D_{ini} and D_{target}. The latter one contained segments with a steady emotional state of the user during 50 or less video frames. Using the dataset D_{ini} for the initial contexts reduces the need in the labelled target context data D_{target} but might hinder the adaptation in the case of significant differences between the initial and target contexts.

(2) Algorithm-specific methods to shift a decision boundary: in these cases, already trained classifiers are modified by model-level knowledge transfer methods, specific to the algorithm, conventionally employed for training these types of models. For example, quadratic programming can be employed for modifying the SVM [37, 80, 81] and Expectation-Maximisation can be employed for modifying the HMM [82]. High-level visual concepts, related to TV news videos, were detected in [37, 80, 81] with the classifiers, based on the binary SVM:

$$k = \operatorname{sign}\left[f_{\mathbf{w};c}\left(\mathbf{x}\right) = \sum_{i=1}^{n} w_i \varphi_i\left(\mathbf{x}\right) + c \right] \in \{-1, 1\}, \tag{4}$$

where $f_{\mathbf{w};c}(\mathbf{x})$ is a decision boundary or a separating hyperplane with coefficients $\mathbf{w}$ and offset c in the n-dimensional space of kernel functions $\varphi_i(\mathbf{x})$ or features of the observed vectors of cues, $\mathbf{x}$. Initially, the classifier was trained on a large labelled set $D_{\text{ini}} = \{(k_t, \mathbf{x}_t)\colon t = 1, \ldots, N_{\text{ini}}\}$ of N_{ini} data items from one TV domain. The quadratic programming based training determined the coefficients $\mathbf{w} = [w_1, \ldots, w_n]$ of the hyperplane that separates the set D_{ini} with the largest margin and correctly classifies data points of both classes in the presence of a tolerable fraction γ of errors $\{\varepsilon_t\colon t = 1, \ldots, N_{\text{ini}}\}$:

$$\begin{aligned} \min_{\mathbf{w}} \Bigg\{ &\frac{1}{2}\sum_{i=1}^{n} w_i^2 + \gamma \sum_{t=1}^{N_{\text{ini}}} \varepsilon_t \mid k_t f_{\mathbf{w};c}\left(\mathbf{x}_t\right) \geq 1 - \varepsilon_t;\ \varepsilon_t \geq 0;\ \mathbf{x}_t \\ &\in D_{\text{ini}};\ t = 1, \ldots, N_{\text{ini}} \Bigg\}. \end{aligned} \tag{5}$$

Simultaneously, the initial training specified a subset D_{sup} of N_{sup} support vectors, located just at the margin distance at both sides of the separating plane.

The adaptation to the new domain in [81] was done by adding to the initial classifier $f^*(\mathbf{x})$, learned from D_{ini}, a shift $\Delta^{\circ}_{\mathbf{w};c}(\mathbf{x}) = \sum_{i=1}^{n} w_i \varphi_i(\mathbf{x})$. The latter was learned in [81] from the labelled new dataset D_{target} of the size N_{target} with due account of the obtained complete classifier $f_{\mathbf{w}}(x) = f^*(\mathbf{x}) + \Delta^{\circ}_{\mathbf{w}}(\mathbf{x})$ and by using the same optimisation framework as the SVM:

$$\begin{aligned} \min_{\mathbf{w}} \Bigg\{ &\frac{1}{2}\left(\sum_{i=1}^{n} w_i^2\right) + \gamma \sum_{t=1}^{N_{\text{target}}} \varepsilon_t \mid k_t f_{\mathbf{w}}\left(\mathbf{x}_t\right) \geq 1 - \varepsilon_t;\ \varepsilon_t \\ &\geq 0;\ \mathbf{x}_t \in D_{\text{target}};\ t = 1, \ldots, N_{\text{target}} \Bigg\}. \end{aligned} \tag{6}$$

This approach aims at placing the new decision boundary close to the initial boundary $f^*(\mathbf{x}) = 0$, probably, because all the news channels under consideration did not notably differ in this work. In the experiments in detecting 39 contexts only from one to ten explicitly labelled samples per concept were used, and the function-level knowledge transfer achieved nearly the same accuracy as the best amongst the more computationally expensive techniques for building context-specific models (more details are provided in Section 4.5).

In cases of notably different initial and target contexts placing a new decision boundary close to the initial one may hinder the adaptation. Hence in [80] a different SVM-specific adaptation approach was suggested: to retrain the models on the combined dataset $D_{\text{com}} = D_{\text{target}} + D_{\text{sup}}$ of the size $N_{\text{com}} = N_{\text{target}} + N_{\text{sup}}$. This dataset is much smaller than the initial one and contains both the labelled new dataset D_{target} of the size N_{target} and the set D_{sup} of the initial support vectors. An application-dependent measure $s(\mathbf{x})$ of relative similarity of each support vector $\mathbf{x}$ from D_{sup} to the current dataset D_{target} was used to reduce the impact of classification errors for these vectors onto the updated coefficients $\mathbf{w}$:

$$\min_{\mathbf{w}} \left\{ \frac{1}{2}\sum_{i=1}^{n} w_i^2 + \gamma \sum_{t=1}^{N_{\text{com}}} s(\mathbf{x}_t)\,\varepsilon_t \mid k_t f_{\mathbf{w};c}(\mathbf{x}_t) \geq 1 - \varepsilon_t;\ \varepsilon_t \geq 0;\ \mathbf{x}_t \in D_{\text{com}};\ t = 1, \dots, N_{\text{com}} \right\}. \tag{7}$$

The employed similarity measure decreases the influence of the support vectors, being located far from the current dataset, onto the tolerable errors:

$$s(\mathbf{x}_\tau) = \begin{cases} \dfrac{1}{N_{\text{target}}} \displaystyle\sum_{t=1}^{N_{\text{target}}} \exp\left\{-\beta \left|\mathbf{x}_\tau - \mathbf{x}_t\right|^2\right\} & \text{if } \mathbf{x}_\tau \in D_{\text{sup}};\ \tau = N_{\text{target}} + 1, \dots, N_{\text{com}}, \\ 1 & \text{if } \mathbf{x}_\tau \in D_{\text{target}};\ \tau = 1, \dots, N_{\text{target}}. \end{cases} \tag{8}$$

The control parameter $\beta \geq 0$ is selected empirically in order to obtain a reasonably high overall performance (Jiang et al. [80] suggested choosing β based on a series of systematic validation experiments). If β is small, all the vectors from D_{com} almost equally take part in the adaptation; that is, the SVM is simply retrained using all the new labelled data together with all the initial support vectors. Hence small values of β better suit cases of fairly similar initial and target domains. The larger the β, the lesser the influence of those support vectors which are far from the new dataset D_{target}. Therefore, choosing a small parameter for the dissimilar initial and target domains would hinder adaptation, while choice of larger values of β would decrease size of the most influential part of the dataset. To compensate for this decrease, it would be desirable to obtain additional data samples for the target domain.

Which of the available initial models should be adapted to the target context is an important but, due to required notable efforts, rarely explored problem. As an infrequent example, a few ways to select the most appropriate model to adapt are compared in [37, 81] with building a target context model from multiple initial models. The ways to select an initial model include (1) comparisons of data or score distributions; (2) classification accuracy comparisons of each initial model and its ensemble, which should be better, by usual assumption, than any of its members; (3) linear regression-based predictions of the initial models' accuracies in the target domain; and (4) iterative user feedback to evaluate how quickly different adapting models improve with the existing labelled data (a history of fast improvement can provide more labelled samples to the model at the next iteration). The multiple initial models appeared to be beneficial in the tests, but none of the tested schemes outperformed others, or even a randomly chosen adaptation significantly (probably, due to relatively small differences between the contexts).

To circumvent selecting the best candidate among the existing initial models, a so-called general model (i.e., a model trained using merged data for all initial contexts) can be adapted instead. That adapting such a general model is feasible has been confirmed for the HMMs in [75] (described in this section above and also in Section 4.5) and in [82]. In [82] the event-specific models for detection of different meeting events, such as note taking and discussion, were obtained from a general model via maximum likelihood estimates of parameters. These estimates required small amounts of the labelled target event data: less than a minute of audio-visual recording per event.

(3) Although some works employed modification of all model parameters for adaptation (e.g., [78, 81]), *adapting only selected parameters* under assumption that other parameters are shared across allcontexts is more common. Proper selection of parameters to adapt requires certain domain knowledge but notably reduces the need in training data. For example, in the model-level knowledge transfer of HMM often transitional probabilities for the hidden states are shared, while the observational ones are adapted: either these probabilities for a discrete HMM directly, as in recognising the show audience excitement [75], or related probabilistic parameters of continuous state-observation relations, like the emission parameters of the Gaussian mixtures in recognising the meeting events [82]. Feasibility of this approach was demonstrated not only for fairly similar initial and target contexts but also for fairly dissimilar ones [75]. In recommender systems *adapting only selected parameters* approach employed an assumption that generic user interests are valid in all contexts [83]. This approach was tested for two quite different initial and target contexts: the generic user interests were inferred first from TV viewing histories and then tourist attractions of the same types were recommended (e.g., diving, for the users watching TV programmes about water sports).

4.2.3. Data-Level Knowledge Transfer. Unlike model-level knowledge transfer methods, transfer methods on data level require training of a new model from scratch: their main goal is to reduce data acquisition efforts.

(1) Error weighting: training the models on a merged dataset, but weighing classification errors on the initial and target contexts differently. This approach suits quite similar contexts and requires either domain knowledge or additional data for defining the weights: too small weights of the old knowledge nearly ignore it, while too high ones hinder proper adaptation. The NN-based affect recogniser in [78] (described in Section 4.2.2) adapted model parameters by minimising a weighted sum of errors on the old and new data. An SVM-based multimedia analysis system [80] (described in Section 4.2.2) was trained on a dataset, containing both labelled target-domain data and support vectors of the initial model, weighed according to their distances from the new training samples (the larger distances are penalised).

(2) The use of model parameters as training data: the learned parameters of an initial model are added to the training data in an algorithm-specific way. In particular, the above SVM-based system [80] for multimedia analysis is adapted by training on the dataset, containing the target-domain data and the support vectors of the initial model.

4.2.4. Lazy Classifiers. For nontrained classifiers the following knowledge transfer approaches were proposed.

(1) Vector modification: many systems store user preferences in a form of vectors, where each element denotes influence of a certain modality on a final result, for example, importance of a certain image feature for the current query and degree of user liking/disliking of a certain interaction modality, movie genre, and so forth. Prediction of a target context vector $\mathbf{S}_i^{\text{target}}$ can be done by shifting vector $\mathbf{S}_i^{\text{ini}}$, obtained for some initial context: $\mathbf{S}_i^{\text{target}} = \mathbf{S}_i^{\text{ini}} + \Delta_i$, where the shift Δ_i of the initial preference vector can be the average difference between the preferences of other users for the two contexts [47], or the classification error minimiser found by differential evolutionary optimisation [84]. Usually, a preference vector of whatever available initial context is modified, and all its elements are modified. This approach requires training data of both initial and target contexts, but fairly little data may suffice: preferences of 20 users served as training data in [84] and preferences of 33 users in [47]. In spite of its simplicity, this approach was successful in predicting user preferences for predefined contexts regarding interface modalities [84] and regarding tourist attractions [47]. In the latter work, variety of usage contexts was fairly large, for example, weather, budget, travel goal, and travel companion.

(2) Modifying a similarity measure: to increase the classification accuracy, the most common linear measures, $\sum_{i=1}^{n} w_i \mathbf{x}_i$, are adapted to different contexts by heuristic or evolutionary optimisation of the weights, w_i, of cues, modalities, or samples, $\mathbf{x}_i$, observed for these contexts. Heuristics is usually based on domain knowledge, while use of evolutionary algorithms usually requires both the initial and the target context data but allows dealing with both similar and dissimilar contexts without estimating the context similarities. Assigning different weights to the same observations for different contexts relaxes the similarity assumption of the CF but provides for no case when this assumption completely fails due to significant differences between initial and target contexts.

To adapt to nondefinable computational factors (different databases of movie ratings) in CF-based recommenders, the work [51] proposed several ways to account for correlations between the contexts in the user similarity measure. Adaptation to nondefinable computational factors (different databases of movie ratings) with optimisation algorithms was successfully employed in [62]. This result demonstrated that *modifying a similarity measure* by an optimisation algorithm does not require notable efforts from each user: in recommender systems the users provide as many ratings as they want, but one user rarely provides many ratings. Adaptation to both definable and nondefinable contexts (individual and group interaction with two fairly different applications: cooking and car servicing assistance) with optimisation algorithms was successfully employed also in [84], where very little training data was used: interaction preferences of 20 users only.

(3) Target context-specific combinations of cues, obtained in other contexts: the cues, obtained for one or more initial contexts, can be combined in different ways, being specific to the target context: the weighted average, voting, rule-based heuristics, and so forth. One rather simple combination scheme for the recommenders, called "post-filtering" [43], uses context-independent recommendation methods, just as prefiltering. But unlike prefiltering, postfiltering gives recommendations on the basis of data, acquired in all contexts, and then either filters or reranks these recommendations for the target context. For example, music recommendations can be provided using all available data and then reranked in line with a historical context, namely, the last songs just listened by the user [85]. Which technique to choose depends on a system goal, as comparison of the prefiltering and postfiltering techniques did not result in a clear winner with respect to different evaluation metrics [86].

Adaptation to social context (group use of an application) often utilises user preferences (cues) obtained in context of individual use of an application. Individual preferences of group members can be combined in various ways [20, 44, 45, 48, 49], but these ways often fail to adapt to heterogeneous groups. A data-adaptive way is to learn, for example, with an evolutionary optimisation, to what extent different input cues may influence the classification in various contexts to find the dominant cues. For example, influences of the group members on group ratings can be learned with a genetic algorithm, using data collected during group and subgroups sessions [87]. However, because collecting the subgroups ratings takes rather long time, this learning was tested on simulated data only [87]. To avoid data collection, designer-defined rules can be used for estimating social dominance and influence of each family member on group ratings, for example, based on age, social role (father, mother, etc.), and income [88]. But due to individual differences and cultural diversity the universal (i.e., suitable for any family) rules are difficult to define.

All the above model-level knowledge transfer scenarios, for example, adapting only selected parameters of the models, can be combined with other model modification methods and various methods to build effective training datasets.

4.3. Ensembles. Lightweight adaptation of ensembles can be performed by selecting the most appropriate base classifier(s) for each context and/or ways to combine their outputs. Context need not be recognised in this approach: ways to select or combine base classifiers depend on their performances in the target context. The ensembles differ basically in two aspects:

(1) Whether outputs of all base classifiers are combined or the best classifier is selected and also, in the latter case, how the base classifiers are selected for each test sample, say, by accuracies on all training data or achieved only for similar samples.

(2) What methods are used to update the ensemble at runtime, for example, only the classifier selection and/or combination rules and/or parameters of the base classifiers can be adapted; and/or some new classifiers can be added, while the underperforming ones can be removed.

Selecting the classifier is the most lightweight option, but it does not accommodate new knowledge easily, whereas updating the base classifiers corrupts the old knowledge. Updating classifier pools keeps both the old and the new knowledge, but optimisation of such pools requires additional training data. Naturally, all these approaches can be combined.

The ensembles differ also by the data usage:

(i) *Context-specific ensembles* train and evaluate their members and selection/combination methods on the target context data. Lazy members search for similar cases only within the target context. This approach suits both similar and dissimilar contexts. Such ensemble may require more data or time to train and test the base classifiers compared to training *context-specific models* (described in Section 4.2.1) but ensures the more accurate adaptation if the base classifiers are diverse enough.

(ii) *Mixed data ensembles* train their members and/or selection/combination methods on both the initial and the target context data and evaluate them on the target context. Lazy members may search within the different contexts or employ different search techniques. Such ensembles significantly reduce the data collection efforts, but their ability to handle both similar and dissimilar contexts depends on specific capabilities of the ensemble members.

Context-specific ensembles usually employ combination of the members' outputs. In the mixed data ensembles, if different members are trained on the data of different contexts, such combinations may work for sufficiently similar contexts. For example, recommender systems often deal with a concept drift (change of the users' interests over time) this way, but then the base classifiers are usually completely retrained at runtime [89, 90]. However, in the mixed data ensembles classifier selection is more appropriate than classifier combination because in the former case the ensemble does not fail if just one member suits well the target context, whereas in the latter case the ensemble fails if the majority of the members fail. The selection of the best member is a good option in the context-specific ensembles, too, because it can be more accurate than classifier combination if trained well [64].

4.3.1. Combination-Based Ensembles Using Context-Specific Data. (1) *Diversity-based ensembles*: training base classifiers on different data chunks or employing different classification algorithms. This is a well-known way to increase the accuracy in conventional ensembles. This approach was used in a multimedia retrieval system [40] to adapt to different user queries: an ensemble of the SVMs was trained on the target context data in such a way that each SVM was trained on all labelled positive examples and negative examples, randomly selected from the unlabelled data (so that different training datasets include different negative examples). Then the ensemble was used to remove wrongly labelled samples: each SVM classified all positive examples, and those out of them, classified negatively by all the SVMs, were considered wrongly labelled. Then a new ensemble was trained on the reduced training dataset, and the final classification combined the outputs of all the SVMs because the ensembles are generally robust to noise. In the tests only five positive training samples for each query were needed, and the training times were short [40]. But the latter times may notably increase for the larger training sets.

This approach was employed also in TV recommender system in [52] to account for differences in viewing habits of different families. Ensemble contained SVM and CBR (case-based reasoning) classifiers, and each classifier employed as features contextual parameters: time and IDs of family members, watching TV. In the tests on 5 months real-life viewing data of 20 families ensemble achieved higher accuracy than any of its members due to their diversity: SVM was less sensitive to nondiscriminative input cues (e.g., in some families choice of programmes depended on time more notably than on presence of family members), whereas CBR better adapted to peculiarities (e.g., cases when choices of {A, B, C} subset of family members notably differed from choices of {A, B}, {B, C}, and {A, C} subsets). This result shows that accuracy of context adaptation may be increased by employing base classifiers with different degrees of sensitivity to nondiscriminative input cues and capable of building decision boundary both locally (as CBR does) and globally (as SVM).

(2) In the *factor ensembles* each base classifier models a certain context-independent factor affecting the final classification result, the factors' weights (in the weighted average, voting, or other fusion methods) being adapted to the context in order to increase the accuracy on the target context data [52, 55, 91]. The training datasets can be quite small if the number of these factors is small. For example, the retrieval of sport videos in [91] used a combination of

several generic attention models, such as those based on camera motion and object detection. The models' weights were adapted to each user based on his/her feedback. The recommenders can also employ this approach; for example, the base classifiers for TV recommenders can be trained on different programme attributes (a genre, a channel, etc.) [52, 55], and the combination of their outputs can be adapted using the feedback data.

4.3.2. Combination-Based Ensembles Using Mixed Data. Mixed data ensembles have been employed for adaptation to personal differences in expressing pain. Chen et al. [92] suggested a method to *optimise a pool of classifiers*, *trained on data for different contexts*: first, for each person in a training dataset a separate ensemble is trained with AdaBoost algorithm to optimise pain recognition rate of this person. All resulting base classifiers of all subjects then constitute a classifier pool, and adaptation to a new person is performed by optimising weights of classifiers in this pool, also by AdaBoost. Optimisation aims at minimising error rate for the target person and is very quick because new base classifiers are not trained at this stage (only existing ones are combined). In the tests the proposed ensemble adaptation took 0.16 minutes per subject on average, and fairly small number of labelled samples per target person (from 25 to 50 samples) allowed achieving notably higher accuracy than that of a generic model. Unfortunately the paper does not explain whether ways to express pain differed significantly between the test subjects and how the proposed method dealt with the persons, most different from the others.

4.3.3. Selection-Based Ensembles. (1) Base classifiers in an *ensemble of generalisers* are trained on the datasets that generalise the target context in different ways, for example, from a nearly exact match up to a mixture of all contexts, and lazy classifiers employ different prefiltering ways to generalise the context [46]. None of these cases requires large datasets for the target context. Success of handling of significantly different contexts depends on the database coverage. The CF-based recommender in [46] included both a general context-independent user profile and content-dependent ones, created using different schemes of generalising the context: for example, for predicting user preferences for a particular "Tuesday" context ranks, acquired on other workdays, were also used. Tests demonstrated the feasibility of including the general profile in such ensembles: it was selected in more than 50% of cases, due to either too weak dependences of the user preferences on some contexts or too small training datasets for these contexts.

(2) Unlike the majority of other ensemble types, employing pattern recognition methods as base classifiers, *knowledge transfer ensembles* employ as base classifiers different strategies to transfer knowledge between the initial and target contexts, for example, mappings between their input cues, or mappings between their output classification results, or mappings between their feedback utilisation models. Such ensembles require both initial and target context data. An ensemble of three relevance feedback techniques for image retrieval in [68, 93] adapted to each query by selecting the most appropriate technique or a combination of the two top-ranked techniques via either reinforcement learning [68] or particle swarm optimisation (PSO) [93]. The training data for each image retrieval query is never large.

A knowledge transfer ensemble in [31] predicted user interface preferences for new applications, screen sizes, newly assembled user groups, and so forth. Each ensemble member modelled a certain human strategy for a new situation: for example, to behave in a target context just as in the initial one; or as the majority of other users in the target situation irrespectively to the initial one; or as those among other users, which behaved similarly in the initial context. The first strategy suits similar initial and target contexts; the second strategy suits the target contexts, strongly affecting the users' preferences; and the third strategy transfers users' similarity. The pool of group strategies included also a few other behaviours, for example, the majority voting. The prediction was based on the known preferences of (1) a target user or group for a single initial context and (2) from 9 to 43 other users or groups for the same initial and target contexts. In the tests with three different applications (a cooking assistant, a car servicing assistant, and a recipe recommender) the interface preferences for each application were predicted based on knowledge regarding either very similar initial context or very different one. For example, preferences for interacting with the recipe recommender were predicted based on the known preferences for significantly different interface of the car servicing assistant.

Which transfer strategy best suits the current context transition and each interface element was estimated from data of user communities. For example, early technology adopters may use a broad range of interaction modalities in many applications, whereas similar attitudes towards healthy lifestyle may lead to similar preferences regarding wellness tips in shopping and cooking aids. However, similar attitudes towards technology adoption do not necessarily imply similar attitudes towards healthy lifestyle. When user similarity is preserved across the contexts and when other strategies are more appropriate follow from the user community data. Required amounts of training data depend on the ensemble size. No domain knowledge is required, provided that users in the database have similar culture: otherwise, their behaviour in new contexts could be too inconsistent to rely on the best ensemble member. This approach may fail also if ensemble members do not cover a sufficient range of the knowledge transfer strategies. On the other hand, it allows adapting to any new situation if different users use similar tags for similar situations, and predefining primary data types is not required provided that some ensemble members can handle new data types. In the tests the ensemble successfully handled transitions between similar and different initial and target contexts and outperformed each of its members.

(3) Base classifiers of the *sample-selecting ensembles* are not pattern recognition methods either: they model different strategies of choosing training samples, for example, between the classes or between the samples of various contexts. Such ensembles may be context-specific or use mixed data and may handle both similar and dissimilar contexts. For example, the aforementioned model-level knowledge transfer in [37,

81] allows for only small shifts of the SVM decision boundary (described in Section 4.2.2). This constraint may hinder the adaptation if the initial and target contexts have significant differences. An alternative approach to SVM adaptation for video concept detection in [94] is to train a model on a dataset, containing the labelled samples from both the source and the target domains, and to select from two strategies of choosing training samples: either near to the decision boundary of the initial model or from most unlikely ones to come from the initial domain data distribution (found by conventional kernel density estimation). The former and latter strategies better suit similar and dissimilar initial and target domains, respectively. To avoid the domain similarity estimation, the samples in [94] were selected as follows: if more samples of a certain type got positive labels at current iteration, a larger proportion of the samples of this type was chosen at the next iteration. The required amounts of the training data depend on task complexity. This approach outperformed the approaches of [37, 80, 81] (described in Section 4.2.2) in experiments on detecting 36 concepts in two rather different video collections. But the adaptation was not very quick, as it required ten iterations.

Originally, the ensembles of sample-selecting strategies were proposed for active learning, that is, for selecting samples from unlabelled data. As it is easier for the users to choose by themselves the samples to annotate, an obvious modification, which can be suggested, is to present more samples than necessary and allow the users to choose and annotate only a portion of these samples. One more suggestion is to test such ensembles on labelled samples from different initial contexts: this may help to find which contexts are most similar to the target context.

(4) *Stacked ensembles* augment a pool of the base classifiers with a pool of their selection and combination strategies. Often, the choice of the selection or combination strategy depends on the training data size and may require data of target context only. The TV recommender in [90] split the data into different time windows and used two decision-making strategies on top of the base classifiers: the members' outputs were combined by voting in the beginning of the time window, whereas in the end the best base classifier was selected. The mixed data ensembles can be also stacked. An ensemble of the two simplest knowledge transfer strategies for interface adaptation in [31] was used when the target context data contained less than 15 samples, more intelligent strategies being added when the dataset size increased. This stacking was used for incremental learning of a new context because too small datasets make selections of a winner unreliable if the ensemble includes many strategies: the required amount of the training data depends on the ensemble size.

(5) *Dynamic selection* of a base classifier can be used in both context-specific and mixed data ensembles. Unlike "static selection" of the most accurate classifier for all data samples, the "dynamic selection" chooses for each test sample a base classifier that achieved the highest accuracy for the similar data samples. The conventional dynamic selection is based on estimating the similarity between test and training samples using their input features [95], as, for example, for the adaptation to different noise conditions in [96]. To model buying behaviour, the best ensemble member is selected in [97] for each product based on its features, such as promotions and dependency of sales on season.

Because the input features similarity is not necessarily preserved across different contexts, a nonconventional dynamic selection approach, proposed for adapting HMM-based systems in [98], is to compute output scores of all ensemble members for a test sample and to compare these scores with the training samples' scores. The ensemble is trained incrementally by adding new classifiers to and removing the least frequently used ones from a pool. However, learning how to select the best subset for each sample requires a 20–25% larger training dataset, comparing with the data needed for training the base classifiers. The overall need in training data depends on the ensemble size and task complexity.

Two other nonconventional approaches, proposed in [57, 68], select either the most appropriate relevance feedback technique for each query and image class by reinforcement learning in an ensemble of such techniques or the most appropriate classifier to handle missing data in affect recognition with a cascade of classifiers, respectively. In the latter case, social rules or personal differences in expressing emotions may exclude certain behavioural cues; for example, needs or habits to be silent result in the absent audio cues. To take the missing data into account, accuracies of all the classifiers are evaluated on training data and stored for each class: the most accurate classifier becomes a "specialist" for this class; the next by accuracy classifier becomes a "second best specialist" for this class, and so forth. The classes are ordered from the most to the least difficult for classification. At the fusion stage a sample is sent initially to the specialist for the most difficult class. If this specialist has classified the sample, the process is terminated to prevent classifying too many samples into the dominant class. Otherwise, the sample is sent to the next specialists until it is classified. To handle the missing data, the sample is sent to the "second best specialist" if the "specialist" for the corresponding class requires the missing modality.

4.4. Context as a Feature. Single and multiple classifiers, using context cues as latent factors, network nodes, or input features, constitute the "*context as a feature*" group. This approach is most feasible in cases when the context cues are discriminative features. The second and third cases also require automatic context recognition. Lazy methods often include context descriptors in a similarity measure. The adaptation can be to either exactly the same context factors as included into the classification model or higher-level situations, described by a set of fine-grain parameters. The adaptation to social rules in different groups using group members' identities or roles as features exemplifies the latter case. Context cues used inside the model increase its complexity and thus the need in training data. Data collection efforts depend also on whether training datasets are context-specific or mixed.

(i) *Context-specific models* are trained on the target context data and suit well cases when user and system behaviour depend on certain fine context parameters, but the dependencies vary for different rough situational strata: for example, dependency of buying behaviour on time of the year differs in different cultures. Therefore, the time context can be a feature, but learning separate models for coarse situations, such as different cultures, is more feasible.

(ii) *Mixed data models* are trained on the data from the target and other contexts or obtained by modifying parameters of models for the initial context. These models are often used for adapting to exactly the same context factors as included in the model. Training a single model for fairly broad ranges of context values allows consistent modelling of context dependency, but accommodating the previously unseen context values would require complete retraining.

Training these models may use either the approaches in Section 4.2 or described below additional four ones, which can be applied to both the context-specific and the mixed data models (the required amounts of training data depend in these cases predominantly on the model complexity).

(1) *Embedding contextual parameters as additional nodes* into graphical models is used in affect recognisers, recommenders, and other systems. A dynamic Bayesian network for drivers' emotion recognition in [99] included additional nodes, taking predefined discrete values to represent both the environmental context (complex versus simple road situation) and the user characteristics (skills, physical condition, and mental state). This allowed to interpret the video cues (e.g., high versus low eyelid position and high versus low gaze fixation) and audio cues (answers to questions) in a context- and user-dependent manner. Data collection was avoided by specifying network parameters by hand. For photo annotation, such nodes can be time, location, and camera parameters, for example, flash [100] or clothing detection, presence of other persons in photos, and demographic statistics for estimating a probability that a person of a certain age and gender has a certain name [101]. In the recommenders the context factors are used also as latent variables, for example, predefined purchase goals as a latent cue in Bayesian networks [102].

(2) *Using historical contexts as nodes in graphical models*: classification of a current data portion in multimedia analysis may depend on the classification results for the previous portions. For example, replays in sport videos usually follow goals but not vice versa. This kind of temporal context can be modelled by HMM, Bayesian networks, or correlation-based graphs [103–105]. Similarly, in affect recognition tasks the past emotional states are often used as nodes, for example, in HMM [75, 106] and so-called Long Short-Term Memory neural networks (LSTM) [106, 107].

(3) *Using contextual parameters as input features*, for example, in the support vector machine (SVM), case-based reasoning (CBR), multilayer perceptron (MLP), naïve Bayes (NB), and decision tree (DT) classifiers: in affective computing a past emotional state served as an additional input to the SVM [75, 108] and AdaBoost [109]. The emotion recogniser for a spoken dialog system [110] employed as the context predefined past events, such as dialogue acts (e.g., repetition and rephrasing, needed when a user cannot be understood immediately) and lexical expressions (e.g., the user's words "no, I said..." may imply correcting a system's mistake that does not make the user happier). Probabilities of emotional categories, associated with these events, were combined with probabilities estimated from acoustic and prosodic features in two stages, employing each a nontrainable fusion, for example, the voting, average, or product of the probabilities. Similarly, dialog acts served as context for differentiating between "doubtful" and "bored" user states in [111].

Time and other predefined fine context descriptors can serve as features in the TV recommenders, learning from long-term interaction histories, but the recommendations are actually adapted to indefinable coarse situations, such as differences between the personal and family cultures. Recommendations for individuals in [112] used a day of week, time of day, user location, and device contexts as inputs to the CBR, MLP, NB, and DT. Recommendations for families in [52, 113] used time and personalities of family members as inputs to the SVM and CBR. Although the "presence of family members" was here a predefined context type, sets of family members were not predefined and varied between the families, which were of different sizes and with or without children. Such learning of the group preferences by observing choices, made by the group members together and separately, respects group practices rather than enforces a practice chosen by a system designer, as was done, for example, in [88] (described in Section 4.2.4). This approach requires no domain knowledge either, but it needs data: for example, achieving reasonable recommendation accuracies in [113] required training data collection during nearly one month. A way to shorten data collection time is to adapt to selected characteristics of users instead of their identities: for example, in [53] "big five" personality traits of individuals served as input features to a neural network, trained to adapt interaction style (e.g., dialog-based and browsing). This approach requires additional data to obtain personality traits, though: in [53] these traits were obtained via analysis of posts, written by the test subjects in social networks.

(4) *Including contextual similarity into a distance measure* is common for context modelling in the CF-based recommenders: for example, distances between the users' ratings can be weighed by similarities between predefined contexts (utilitarian versus hedonic users' needs, a day of week, and time of day), in which these ratings were provided [114]. Including context in a distance measure can also help to deal with changes in the users' interests over time: for example, an order of items' consumption and difference between consumption times [115].

4.5. Comparing the Adaptation Approaches. Experiments with more than one adaptation framework are reviewed in brief below. Unfortunately, no consistent experimental comparisons of the different approaches could be found in the literature. Moreover, while in some cases only the lightweight adaptation led to the desired functionality [31], the fine adaptation required in other cases so significant data collection efforts that even the application developers did not want to meet with these problems [75].

The adaptation is more lightweight if feature selection is not performed for each context. Using the same features was compared with the feature selection in [116] for object categorisation by fully supervised learning vector quantisation method. The data were challenging as it contained images with rotated by various angles objects of different shapes and colours. Using the same features for different object categories did not notably lower the accuracy, comparing with the feature selection. Therefore, a good initial choice of the features may allow for learning new categories even without extensive feature selection for each new data portion.

In other comparative experiments, discussed below, feature selection was not performed for each context. The ensemble-based adaptation was compared in [37, 81] with three methods to build context-specific SVMs for detecting concepts in the TV data. The SVMs, trained separately on the data from different contexts and combined using a weighed sum with weights stating importance of the new context, formed the ensemble. The context-specific SVMs were built in the following ways: (i) by the model-level knowledge transfer (described in Section 4.2.2); (ii) by training on the merged data from the old and new contexts, and (iii) by training on the data of the target context only.

In the tests on the data from 13 TV channels, the model-level knowledge transfer was close by the average accuracy to the models, trained on the merged data, but the training was up to 15 times faster. The purely context-specific models were less accurate due to a small number (from one to ten per concept) of positive training samples. The ensemble accuracy was close to that of the knowledge transfer if the target context weight was not high. Additionally, a semisupervised SVM, trained on both the labelled and the unlabelled target context data, but with no knowledge of initial contexts, was evaluated in [37] and appeared to be significantly less accurate than the model-level knowledge transfer.

Several ways to build a context-specific HMM for recognising excitement of show audience from an audio-visual stream were compared in [75] (these ways are described in Sections 4.2.1 and 4.2.2): (i) cascaded training on the target context data; (ii) model-level knowledge transfer (either the context-specific HMM, trained for some other context, or a general HMM was adapted to the target context, using its labelled data), and (iii) unsupervised training of the conventional HMM on the unlabelled target context data. To demonstrate benefits of the context adaptation, the context-specific models were also compared with a general HMM, built either by unsupervised or by cascaded training on the data of all contexts.

The cascaded training on the target context gave the highest accuracy in these tests. The general HMM, trained in the cascaded way, was also more accurate than the conventional HMM. Although feasibility of unsupervised pretraining of deep neural networks was demonstrated in [74], in training of less deep architectures usually all the available data is utilised at once. According to [26, 75], the cascaded training may be beneficial in nondeep architectures, too.

The model-level knowledge transfer achieved slightly lower accuracy than the cascaded training. As regarding the choice of a model to modify, various initial context models were quite similar by their accuracies after adapting to the target contexts (probably, due to the notably different contexts). The adapted general model was slightly more accurate. The conventional context-specific and general HMMs achieved significantly lower accuracies than the model-level knowledge transfer. In addition, accuracy of fully supervised SVMs, trained on the labelled target context data, was also presented in [75]. In the tests the SVM and HMM accuracies were similar to each other for a large number of the annotated samples provided, but with only 25 labelled samples per context the SVM became much less accurate than the HMM.

The cascaded architectures were beneficial in other applications, too. Updates of final rather than initial stages of a cascaded system in [32] were more efficient for speech processing. A single model, trained for the TV recommender in [55] on several attributes of TV programmes, adapted to individual tastes less accurately than a classifier ensemble. Each member of the latter was trained on a single programme attribute (such as “genre”), and the models’ outputs got different weights for the different users. The hierarchical and nonhierarchical methods of learning interconcept relations to detect concepts in images were compared in [76, 103]. The hierarchical approach built first binary classifiers for each concept separately and then adapted their relations to contexts [76] (more details are provided in Section 4.2.1), while the nonhierarchical learning built such a relational model directly from the low-level features [103]. In the tests both approaches achieved fairly similar accuracies, but the hierarchical adaptation was significantly faster.

Different ways of knowledge transfer have been compared in a study into pain recognition ([92] (described in Section 4.3.2)). The lightweight adaptation of a combination-based ensemble was compared with two approaches: (1) training of a person-specific model using data of the target person only and (2) training of a person-specific model using data of all test subjects. The tests proved that lightweight adaptation is notably faster: it took 0.16 minutes per person on average, while training on the data of the target person took 2.6 minutes and training on the data of all subjects took 14.3 minutes per subject on average. The ensemble was also notably more accurate than a person-specific model, trained on data of the target person only, in cases when the number of labelled training samples per target person was fairly small (from 10 to 50 samples). The accuracies of the ensemble and of the person-specific model, trained on data of all subjects, were fairly similar when number of labelled training samples per target person ranged from 10 to 25, but the ensemble was notably more accurate when training dataset per target person included 50 and more samples.

Different knowledge transfer strategies have been compared in [31, 84] in a study into user interface adaptation. The ensemble of several simple strategies was used in [31] (described in Section 4.3.2), and two popular alternative approaches were tested in [84] on a part of the dataset used in [31]. One of them transfers the knowledge by adding or subtracting a shift vector from the vectors of preferences for each user (or user group), similarly to [47]. Comparing to other approaches, its accuracy in the tests was considerably lower. The second approach exploited one of the most common adaptation methods: modifying a similarity measure by an optimisation algorithm to minimise the prediction error for all nontarget users. This approach appeared to be fairly accurate and outperformed other approaches in cases when the users' preferences in the initial context were similar to each other but differed significantly in the target context. However, this approach required much longer computations, and its average accuracy was similar to that of the ensemble in [31].

The use of the target context data only versus the mixed data in the ensemble of relevance feedback strategies for image retrieval was explored in [68] (described in Section 4.3.3.). The target context included interactions of a current query, and the mixed data contained interactions of multiple retrieval sessions of multiple users. In the tests the mixed data greatly increased the precision of initial results (images retrieved before using the relevance feedback). The precision after the first relevance feedback iteration was also improved, but less notably, whereas after the second iteration the precision with the mixed data was only 2.5% higher than that with only the current query data, and the time gain was insignificant, too.

Adaptation to uncontrolled environments has to deal with possibilities that not all input modalities are always available. The cases of missing or poor quality data samples were studied in biometrics, because an important goal for next generation biometrics is to increase user convenience [117, 118]: for example, users may dislike certain biometric modalities or be incapable of providing data due to trauma. The missing data can be handled by training models for different combinations of modalities and selecting an appropriate model for each combination [119, 120] or by applying generic methods to fuse all modalities in presence of missing data, such as imputation of the missing samples or modification of the fusion algorithm [121]. Suitability of generic methods, however, depends on missingness mechanism, that is, whether data are missing at random or not [121]. For example, imputation methods do not fit well the "data are missing not at random" scenario [121], whereas "data are missing at random" assumption is infeasible when context influences availability of modalities (e.g., if etiquette requires silence, voice data will be unavailable). Besides, studies, comparing the model selection with imputation and the modified SVM fusion [119, 122], demonstrated that the model selection was more accurate on the average. Samples of various qualities in biometric systems can be handled also by model selection (training aseparate model for each quality cluster) or using the sample quality as an additional input feature. According to the tests in [123], the former approach is more accurate, especially if the number of the quality measures increases. These results suggest that nondiscriminative contextual parameters, like the sample quality factor, should not be used as features.

Fine context descriptors, such as time, can be discriminative factors in the recommender systems. Use of context descriptors as features in the trained (SVM) and lazy (CBR) classifiers in the TV recommenders was compared in [52, 113] (methods are described in Section 4.4). In the tests on real-life TV viewing histories of 20 families both classifiers achieved quite similar average accuracies.

Table 1 presents most important characteristics of the reviewed lightweight adaptation approaches. These features can have positive or negative influence on adaptation, depending on the task at hand. For example, ability of the adaptation approach to use unlabelled data for the target context is a positive feature because it decreases the need in the labelled data. On the other hand, such approaches cannot be employed in applications where unlabelled data cannot be obtained. Similarly, most lightweight approaches suit well cases when only very little datasets for the target context can be obtained, but their accuracies are usually lower than that of the less lightweight methods. ("Most lightweight" and other characteristics of adaptation approaches in Table 1 refer to runtime training in case when a new context emerges.)

5. Lightweight Adaptation Design

The above overview allows us to identify important application characteristics, influencing the adaptation design, and suggest choices of the adaptation methods, which are most likely to work for different application requirements.

5.1. Choices and Factors Influencing Design Decisions. Let the same types of data and context cues, although not necessarily exactly the same sets of cues, be used in all contexts. Then the most important design choices are the interaction, adaptation, and data usage types. The interaction can be implicit or explicit or can combine both. The adaptation types are model selection, classifier ensemble, or using context as a feature. The data usage type is the knowledge transfer type and level: either no data, or a dataset, acquired in other situations, or a model trained on a dataset, acquired in other situations, can be employed.

These choices strongly influence choice of a reasoning method and choice of runtime training and supervision types. The reasoning method can be either lazy or trained graph-based (e.g., HMMs and Bayesian networks) or non-graph-based (e.g., neural networks and SVMs). Runtime training can be performed by either conventional standard algorithms (e.g., the Baum-Welch one to train HMMs) or custom parameter modifications, including the choice of parameters to modify and/or certain simplifying assumptions, for example, linearization of functions and constraints on parameters' changes. The supervision can be full or partial (unsupervised context adaptation usually ignores peculiarities). All these decisions depend on each other and on task specifics.

The following application characteristics influence the design the most:

(i) *Expected changes in input cues in different situations, namely, in their (1) meaning; (2) availability; (3) influence; and (4) accuracy*: for example, availability

Table 1: Characteristics of the reviewed adaptation approaches. Notations: X: yes; A: depending on a chosen algorithm; for example, training data can be used instead of assumptions about new context and unlabelled data are used only as negative examples; MK: multiclass classification problems only; U: data are vectors of user preferences; base classifier: either ensemble member or the only context-specific model.

	Applicability				Training		Additional data			
Method name	Suits dissimilar contexts	Suits large varieties of problems/algorithms	Most lightweight	Requires little assumptions about new contexts	For training of base classifier(s)	For classifier selection/combination	Uses contextual data	Uses raw primary data of other contexts	Uses models for other contexts	Uses unlabelled data for the target context
					Model selection					
Contextual weighting	X	X	A	A	X	X	A			
Optimising utility function	X				A	A	A			A
Tuning classifiers for small datasets	X			X	X					A
Cascaded training	X	X		X	X					X
Learning context-specific relations between classifier outputs	X	MK	X	X		X				A
Optimising model parameters with evolutionary algorithms	X	X		X	X	X		A	X	
Optimising model parameters with gradient descent	A	X		A	X			A	X	
Algorithm-specific methods to shift a decision boundary	A			X	X			A	X	A
Adapting only selected parameters	X	X	X		X	X		A	X	
Error weighting		X			X	X		X	A	
The use of model parameters as training data	A				X				X	
Vector modification	X	U	X	X	X			X		A
Modifying a similarity measure	X	U	A	A	X			A	A	A
Target context-specific combinations of cues, obtained in other contexts	X	U	A		X	X	A	A		A
					Ensembles					
Factor ensembles	X		X	X		X				A
Diversity-based ensembles	X	X		X	X					A
Optimising a pool of classifiers, trained on data for several contexts		X	X	X		X			X	
Ensemble of generalisers			X			X	X		X	

Table 1: Continued.

	Applicability				Training		Additional data			
Method name	Suits dissimilar contexts	Suits large varieties of problems/algorithms	Most lightweight	Requires little assumptions about new contexts	For training of base classifier(s)	For classifier selection/combination	Uses contextual data	Uses raw primary data of other contexts	Uses models for other contexts	Uses unlabelled data for the target context
Knowledge transfer ensembles	X	X	X	X		X	A	X	A	A
Stacked ensembles	X	X	X			X				
Dynamic selection of base classifiers	X	X	A	X		X	A	A		
Sample-selecting ensembles	X	X		A	X			X	A	A
Context as a feature										
Embedding contextual parameters as additional nodes into graphical models	X				X		X	X		A
Using historical contexts as nodes in graphical models	X				X		A	X		A
Using contextual parameters as input features	X	X		X	X	X	X	X		A
Including contextual similarity into a distance measure	X		X			X	X	X		A

of all cues is not guaranteed in uncontrolled environments; the low-level cues, for example, image features, usually have no meaning to change, whereas high-level behavioural cues may easily change meaning, for example, user's silence after a show is likely to indicate user's disappointment as satisfied users would applaud, whereas silence in the beginning is more likely to indicate neutral mood (just waiting for a show to start).

(ii) *Effects of different interaction types (i.e., costs of data acquisition versus data quality): (1) implicit interaction and (2) explicit interaction*, that is, whether users and classifiers can significantly benefit from explicit human efforts and whether implicit interaction can be reliably interpreted.

(iii) *Variability of situations a system is likely to encounter: (1) stable versus unstable and (2) definable versus undefinable situations*: for example, an in-house TV programme recommender should adapt to any family acquiring it, but only to this particular family, whereas an Internet TV programme recommender service has to adapt to each family using the service. To define several screen types for adapting an UI is an easy task but to specify all situations where humans' behaviour is governed by social rules is much more difficult. Such elusive situations will be called "indefinable" below.

(iv) *Adaptation time*: that is, whether the adaptation must be very quick (e.g., the interface should be adapted just at the moment of launching an application) or can take time or whether a lifelong learning is expected.

Figure 3 shows which adaptation approaches, outlined in Section 4, were suggested for different subtypes of the three most influencing types of application characteristics. Although many of these approaches were used offline, the proposed techniques suit the runtime adaptation, too. Subtypes of the application characteristics are ordered along the corresponding axes by increasing complexity: among changes in input cues, meaning changes are the trickiest, while the simpler changes in the cues' influences are the most common case; among interaction types, the explicit interaction is considered more difficult than the implicit one because the former often results in very small datasets. Among situations types, "many diverse indefinable situations" is the most difficult case. The application characteristics are either taken from the papers reviewed or based on our own assumptions: for example, the accuracy and meaning of low-level image features, such as colour histograms, rarely depend on context. In particular, if a colour histogram is used for distinguishing between cars and houses, its meaning would be database-dependent only if the majority of cars and the majority of houses are red and green, respectively, in one database, while these colours are swapped in another database. However, this is too unlikely in the large databases. Therefore, we assume that the low-level image features can only change their influence on a classification result.

5.2. Interaction Types. As shown in Figure 3, the adaptation to changing input cues' meanings is rarely addressed. Several studies suggest that quick learning of new meanings may be difficult without explicit human supervision. Thus, acquiring explicit interaction data can be suggested for the changes in meanings of the input cues. Explicit efforts are necessary also if interpretation of implicit interaction data depends on context: for example, if just the same user actions may denote a positive feedback in one context and neutrality in another context.

The implicit interaction should be employed, if possible, when quality of primary data cues is rather low. For example, if a certain cue is important for users, but not recognised by a system, accurate models cannot be built despite all the explicit annotation efforts; instead, the latter will only annoy the users. Due to this reason, TV recommenders often employ an implicit feedback: TV programme metadata is rarely detailed.

Otherwise, for choosing between the implicit and explicit interaction we recommend considering, first, how the UI design affects quality of implicit data and, second, to what extent both labelling time and a time interval when the labels are used by the system match the users' goals. For example, if the users click on nearly every link because of insufficient link data, the implicit feedback would be useless. On the other hand, benefits of explicit feedback should last longer than its acquisition: for example, interactive multimedia retrieval systems usually learn from small datasets because users only see how the current query results are improved by their feedback, whereas recommender systems may require users to rank many items and to use these ranks for next year(s). Third, we recommend considering whether users' actions follow their own desires or social rules: in the latter case the implicit feedback may be useless if the users adjust their choices to be polite. This behaviour, however, is less common among loving families and close friends than in groups where some members dominate. To understand whether all group members are equally satisfied is difficult [20, 31], but solutions to this problem remain to be found.

5.3. Adaptation and Data Usage Types. A very quick adaptation can be performed by (i) selecting from a set of models, trained for predefined situations; (ii) using the context as a feature in a model, trained for a broad range of context cues, and (iii) employing selection-based classifier ensembles with nonadaptive base classifiers. Recommendations for choosing adaptation and data usage types for other cases are given in Figure 4. These choices are closely related, but the usages of data of one or several nontarget situations are not differentiated because no sound clues exist in the state-of-the-art research.

Using both labelled and unlabelled target context data could be beneficial when both could be obtained. Benefits of using unlabelled data of the nontarget situations in the lightweight adaptation have not yet been demonstrated. Usually, the decision on whether to use only the target context data or also the labelled data from other situations for adapting to a target context is based on the domain knowledge or data comparisons for these situations (e.g., comparison of data distributions). The latter approach is not

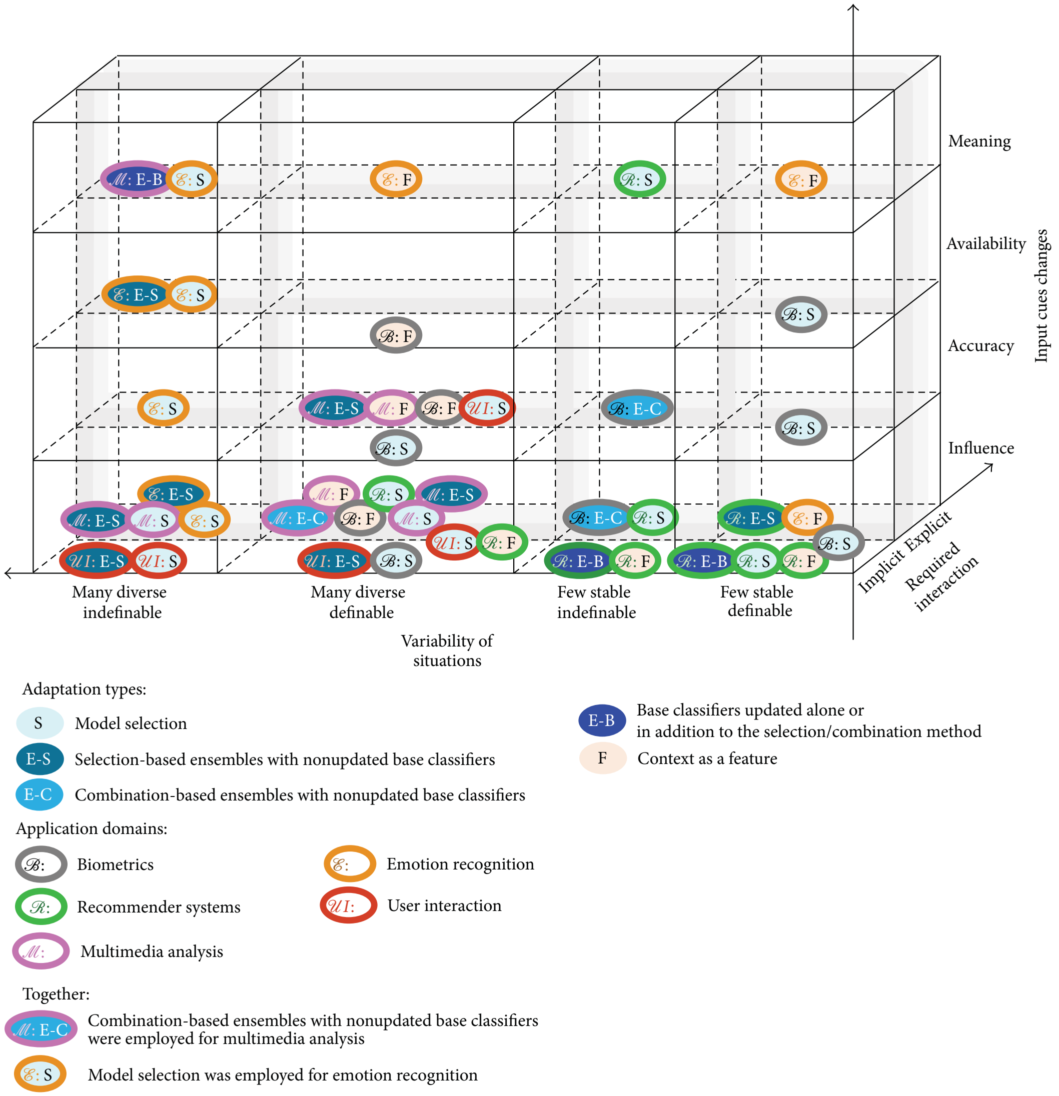


Figure 3: Lightweight adaptation summary (Section 4).

quite lightweight, so we suggest employing simple heuristics instead: the more serious the changes in the input cues are expected, the more dissimilar the situations in question are.

As Figure 3 shows, cases of changes in influence of input cues on adaptation result are best studied. If just such changes are expected and an application will be used in a few fairly stable easy-to-define situations, nearly any adaptation approach from Section 4 can be applied. In cases of changes in meanings of input cues choice of adaptation approaches is rather limited. Training a separate model for each context is one of the most adaptive approaches. It can be applied to a broader range of cases than those having been studied so far and handle primary data cues, emerging at runtime, provided that their types and the use of different types inside the models can be predefined and the models can be trained at runtime. On the other hand, the adaptation to changes in the input cues availability may be reached without training a separate model for each context if the cue availability is treated as a feature in reasoning methods that handle the missing cues without retraining, for example, discrete HMMs, weighted sums, and voting.

Among the above-mentioned approaches to obtain the context-specific models, two generic ones can be emphasised: (i) the *cascaded training* using only the target context data if

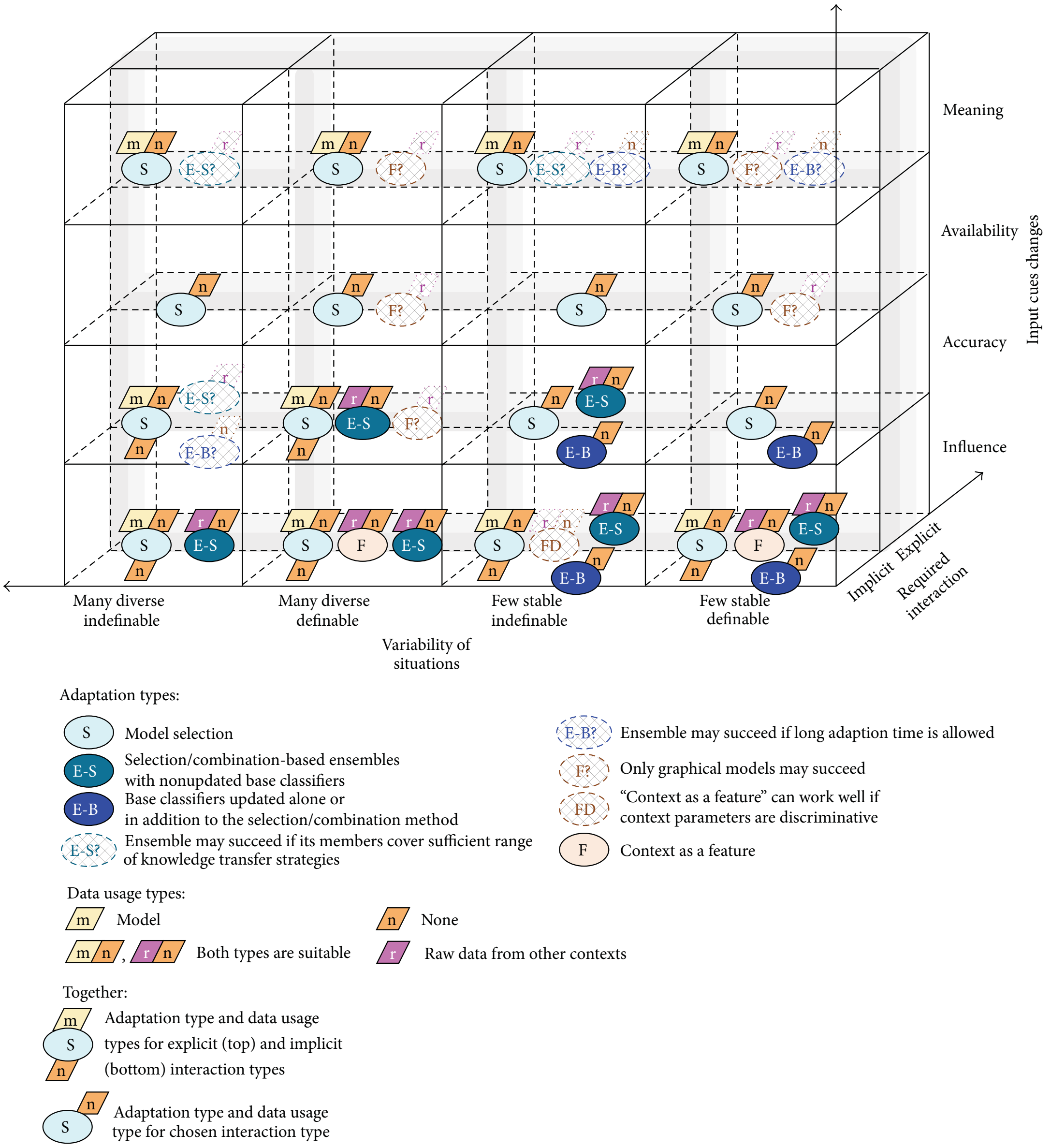

Meaning
Availability
Accuracy
Influence
Input cues changes
Implicit
Explicit
Required interaction
Many diverse indefinable
Many diverse definable
Few stable indefinable
Few stable definable
Variability of situations
Adaptation types:
S Model selection
E-S Selection/combination-based ensembles with nonupdated base classifiers
E-B Base classifiers updated alone or in addition to the selection/combination method
E-S? Ensemble may succeed if its members cover sufficient range of knowledge transfer strategies
E-B? Ensemble may succeed if long adaption time is allowed
F? Only graphical models may succeed
FD "Context as a feature" can work well if context parameters are discriminative
F Context as a feature
Data usage types:
m Model
n None
m n, r n Both types are suitable
r Raw data from other contexts
Together:
Adaptation type and data usage types for explicit (top) and implicit (bottom) interaction types
Adaptation type and data usage type for chosen interaction type

Figure 4: Recommended adaptation and data usage types.

the unlabelled data is available and (ii) the *model parameters' modification by evolutionary algorithms*. Cascaded training was proved beneficial for deep neural networks [74] and MLP [26] (trained conventionally in fully supervised ways) as well as for HMMs (trained conventionally in unsupervised ways) [75]. Evolutionary algorithms are worth considering because they are fast and applicable to various sets of parameters. Furthermore, they require no differentiable penalty function and are not easily stopped by local minima. These algorithms can be employed for both the model-level knowledge transfer (shifting a decision boundary) and the cascaded training (at its last stage). Other learnable models may also benefit from the evolutionary adaptation.

The classifier ensemble, where all members are trained on the target context data, is the most adaptive but is less lightweight than training a separate model for each context.

Accordingly, employing such ensembles to adapt to contexts, emerging at runtime, is feasible mainly when large datasets are acquired naturally in the course of using an application, for example, when implicit interaction data can be collected. In other cases, either classifiers learnable from very small datasets are to be included into the ensembles or other ensemble types should be chosen.

Combining outputs of the base classifiers can be recommended only when all these classifiers are sufficiently generic (e.g., in factor ensembles) or trained on target context data. In other cases the selection-based ensembles should be employed: they can handle both similar and dissimilar contexts if different members are optimised for different contexts [31], and they outperform individual classifiers when amount of training data is not abundant [95].

Unlike the conventional classifier ensembles, built from pattern recognition strategies, the context-adaptive ensembles may include other reasoning strategies. Among the latter, the knowledge transfer strategies are most suitable for cold-start adaptation to newly emerged contexts. Such ensembles can adapt to new contexts without retraining the base classifiers after only a few target context data samples are acquired, provided that the transfer strategies can be defined for a task at hand. On the other hand, the ensemble should not be of too large size because the more the members to be evaluated, the more the data required [31]. The ensembles can also handle the primary data cues, emerging at runtime, if some of their members (e.g., lazy methods) handle such cases. Typical behaviours of individuals and groups [31]; typical weights of input cues; and typical types of relations between input or output cues (yet to be tested) may be included into the knowledge transfer strategies' ensembles. After more target context data is collected, more sophisticated adaptation methods can be employed instead, for example, by stacking the ensembles.

Recommendations on using the context as a feature refer to single classifiers. Single classifiers handle changes in the input cues' influences the best. The graphical models, for example, HMMs and Bayesian networks, adapt to such changes by adjusting the observational or transitional probabilities. The HMMs suit well adaptation to the historical context factors. Using context parameters as features in nongraphical models (e.g., in the SVMs) can be recommended only when the context factors are discriminative. Similarly, model selection and ensembles can also include models, using context descriptors as features.

Regarding the data usage types, training the context-specific models on the merged data of different contexts may hinder the adaptation to notably different contexts. Therefore, the use of the merged data can be recommended only if the chosen adaptation requires such data: for example, for an ensemble of knowledge transfer strategies or for optimising similarity measures. Using trained models of the nontarget contexts can be recommended for faster adaptation. To use the target context data only can be recommended when the training datasets are not too small (e.g., if implicit interaction data or unlabelled data are available) or the applications are supposed to be long-term user companions and thus need to gain the user's trust by avoiding data sharing and reasoning errors.

5.4. Training and Supervision Types. Choices of the training and adaptation types are strongly interdependent. Model-level knowledge transfer and old knowledge preservation usually require custom algorithms; for example, some additional constraints on the model parameters may be added. Custom adaptation, such as with evolutionary algorithms or various reweighting schemes, is usual in selecting and/or combining the ensemble members, too. Standard training is more common in other cases due to its easiness for developers.

The choice of a training supervision type depends strongly on the chosen interaction and data usage types. If no knowledge transfer is used, the training dataset, especially the one obtained via explicit interaction, may be too small for the fully supervised training. If an application allows for acquiring the unlabelled data, the cascaded training should be employed. The conventional semisupervised training cannot be recommended for adapting to contexts, emerging at runtime, as correct modelling assumptions are difficult to make for new contexts and the incorrect ones decrease the accuracy of the semisupervised learning comparing with the use of the labelled data only [26].

6. Conclusions

The lightweight runtime adaptation of classifiers to newly emerging situations, not requiring significant explicit interaction efforts and the detailed domain knowledge, is a new research area with important practical applications for user-centric multimedia analysis and retrieval, automated UI adaptation, recommender systems, and so forth. We reviewed promising adaptation techniques, proposed in several application domains, and provided recommendations on selecting such techniques, based on the identified application characteristics and simple heuristic evaluations of similarities between contexts. Our focus is on adapting class-level and decision-level multimodal fusion under the assumption that inputs to the fusion models (called cues) are provided by the same algorithms in all contexts. Several studies have shown that despite this significant limitation the lightweight adaptation can be close by classification accuracy to more computationally expensive adaptation methods, whereas in many cases it simply has no alternatives to compare with. Moreover, if the full-scale adaptation is possible, it can be performed after the lightweight one, either if the end users did not accept the latter results or when additional data becomes available.

Two main problems of the full-scale adaptation are the extensive use of the domain knowledge in reasoning and the need in the labelled training data. The above lightweight approaches are more data-driven than domain knowledge dependent; that is, they employ sufficiently generic input cues and learn to handle them in context-specific ways, using the target context data. For example, the same user or system behaviour can be assigned to different inner states of a HMM classifier by modifying observational probabilities (in particular, the classifier can be trained to treat a sound of "whistling" as a sign of the user excitement in one context and "nothing special" in another context). Ensembles of methods for knowledge transfer between contexts and data of several contexts allow for learning which transfer method is more appropriate for the initial and target contexts at hand: for example, which relevance feedback strategy better suits the current user query or which types of user preferences strongly depend on contexts. The domain knowledge dependency can be reduced also by employing the reasoning that naturally deals with missing inputs and cues emerging at runtime. For example, graphical models can treat a missing cue as a valid observation; if various types of cues are read in different ways from data logs, it is sufficient to specify how to treat a cue type in reasoning and specifying an exact set of the cues is unnecessary.

At the same time, the need in the labelled data can be reduced by using the unlabelled data in addition to the labelled data; by modifying training algorithms so that very small sets of the labelled data would be sufficient; or by reusing the knowledge regarding multiple contexts, or multiple users. For example, the conventional assumption that users, similar to each other in the past, remain similar also in the current situation does not necessarily hold for significantly different past and current situations. Therefore, the user community data can be used first to check whether users, similar to each other in a certain initial context, remain similar also in a target context. Then, if this would be the case, the data for this initial context can be used in addition to the data of the target context. Otherwise, only target context data should be used.

That the lightweight adaptation is feasible was confirmed in the aforementioned studies. However, trade-offs between the adaptation cost and performance gain, acceptable for the end users, remain an open problem. Moreover, the user acceptance depends not only on the classification accuracy but also on many other factors, such as user companions, screen sizes of interaction devices, general user attitudes to adaptability of personal devices, and reasoning implementation. For example, in the UI adaptation [31] the reasoning employed user community data, and many subjects preferred this way to a more common launching of the applications with settings provided by application designers, because of the higher confidence in their acquaintances. Thus, certain types of knowledge transfer, in addition to reducing the need in data collection for the target situation, may also facilitate the users' trust.

The lightweight adaptation is suitable for various practical problems, including a cold-start adaptation when it is not yet possible to acquire sufficient data for more sophisticated methods. Also, it is beneficial for interactive systems that should respond fast to users' requests, as well as for systems encountering a large variety of the usage situations. The designers of the latter systems would be unable while the end users will be unwilling to invest big efforts in data collection. A full-scale adaptation should not be the only option for the end users of both kinds of systems; it may be more feasible to try the lightweight one first to see whether the users are satisfied with it. Although the lightweight adaptation methods, reviewed in this work, were usually tested on the data of one application domain only, we believe that other domains can also benefit from these studies. For example, the work [2] concluded that pervasive computing applications should employ knowledge transfer learning methods to greater extent to reduce data collection needs. Among the lightweight adaptation methods reviewed one can also find generic enough ones, being useful not only for context adaptation but also for user adaptation and solving other machine learning problems.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] L. Cao, "Domain-driven data mining:challenges and prospects," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 6, pp. 755–769, 2010.

[2] J. Ye, S. Dobson, and S. McKeever, "Situation identification techniques in pervasive computing: a review," *Pervasive and Mobile Computing*, vol. 8, no. 1, pp. 36–66, 2012.

[3] B. Khaleghi, A. Khamis, F. O. Karray, and S. N. Razavi, "Multisensor data fusion: a review of the state-of-the-art," *Information Fusion*, vol. 14, no. 1, pp. 28–44, 2013.

[4] C. Krupitzer, F. M. Roth, S. VanSyckel, G. Schiele, and C. Becker, "A survey on engineering approaches for self-adaptive systems," *Pervasive Mobile Computing*, vol. 17, pp. 184–206, 2015.

[5] J. Lu, V. Behbood, P. Hao, H. Zuo, S. Xue, and G. Zhang, "Transfer learning using computational intelligence: a survey," *Knowledge-Based Systems*, vol. 80, pp. 14–23, 2015.

[6] A. Smirnov, T. Levashova, and N. Shilov, "Patterns for context-based knowledge fusion in decision support systems," *Information Fusion*, vol. 21, no. 1, pp. 114–129, 2015.

[7] L. Snidaro, J. García, and J. Llinas, "Context-based Information Fusion: a survey and discussion," *Information Fusion*, vol. 25, pp. 16–31, 2015.

[8] M. Xu, C. Xu, L. Duan, J. S. Jin, and S. Luo, "Audio keywords generation for sports video analysis," *ACM Transactions on Multimedia Computing, Communications and Applications*, vol. 4, no. 2, article 11, 23 pages, 2008.

[9] W. Hu, N. Xie, L. Li, X. Zeng, and S. Maybank, "A survey on visual content-based video indexing and retrieval," *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, vol. 41, no. 6, pp. 797–819, 2011.

[10] A. Rehman and T. Saba, "Features extraction for soccer video semantic analysis: current achievements and remaining issues," *Artificial Intelligence Review*, vol. 41, no. 3, pp. 451–461, 2014.

[11] P. K. Atrey, M. A. Hossain, A. El Saddik, and M. S. Kankanhalli, "Multimodal fusion for multimedia analysis: a survey," *Multimedia Systems*, vol. 16, no. 6, pp. 345–379, 2010.

[12] C. A. Bhatt and M. S. Kankanhalli, "Multimedia data mining: state of the art and challenges," *Multimedia Tools and Applications*, vol. 51, no. 1, pp. 35–76, 2011.

[13] M. Kotti and F. Paternò, "Speaker-independent emotion recognition exploiting a psychologically-inspired binary cascade classification schema," *International Journal of Speech Technology*, vol. 15, no. 2, pp. 131–150, 2012.

[14] A. Tawari and M. M. Trivedi, "Speech emotion analysis: exploring the role of context," *IEEE Transactions on Multimedia*, vol. 12, no. 6, pp. 502–509, 2010.

[15] L. Nguyen, J.-M. Odobez, and D. Gatica-Perez, "Using self-context for multimodal detection of head nods in face-to-face interactions," in *Proceedings of the 14th ACM International Conference on Multimodal Interaction (ICMI '12)*, pp. 289–292, ACM, October 2012.

[16] C. Shin and W. Woo, "Socially aware TV program recommender for multiple viewers," *IEEE Transactions on Consumer Electronics*, vol. 55, no. 2, pp. 927–932, 2009.

[17] R. Sotelo, Y. Blanco-Fernández, M. López-Nores, A. Gil-Solla, and J. J. Pazos-Arias, "TV program recommendation for groups based on muldimensional TV-anytime classifications," *IEEE Transactions on Consumer Electronics*, vol. 55, no. 1, pp. 248–256, 2009.

[18] J. Y. Xu, H.-I. Chang, C. Chien, W. J. Kaiser, and G. J. Pottie, "Context-driven, prescription-based personal activity classification: methodology, architecture, and end-to-end implementation," *IEEE Journal of Biomedical and Health Informatics*, vol. 18, no. 3, pp. 1015–1025, 2014.

[19] J. Borràs, A. Moreno, and A. Valls, "Intelligent tourism recommender systems: a survey," *Expert Systems with Applications*, vol. 41, no. 16, pp. 7370–7389, 2014.

[20] J. Masthoff and A. Gatt, "In pursuit of satisfaction and the prevention of embarrassment: affective state in group recommender systems," *User Modeling and User-Adapted Interaction*, vol. 16, no. 3-4, pp. 281–319, 2006.

[21] B. Dumas, M. María Solórzano, and B. Signer, "Design guidelines for adaptive multimodal mobile input solutions," in *Proceedings of the 15th International Conference on Human-Computer Interaction with Mobile Devices and Services (Mobile-HCI '13)*, pp. 285–294, ACM, 2013.

[22] C. Evers, R. Kniewel, K. Geihs, and L. Schmidt, "The user in the loop: enabling user participation for self-adaptive applications," *Future Generation Computer Systems*, vol. 34, pp. 110–123, 2014.

[23] A. Vinciarelli, M. Pantic, D. Heylen et al., "Bridging the gap between social animal and unsocial machine: a survey of social signal processing," *IEEE Transactions on Affective Computing*, vol. 3, no. 1, pp. 69–87, 2012.

[24] M. R. Ghorab, D. Zhou, A. O'Connor, and V. Wade, "Personalised information retrieval: survey and classification," *User Modelling and User-Adapted Interaction*, vol. 23, no. 4, pp. 381–443, 2013.

[25] S. T. Shivappa, M. M. Trivedi, and B. D. Rao, "Audiovisual information fusion in human-computer interfaces and intelligent environments: a survey," *Proceedings of the IEEE*, vol. 98, no. 10, pp. 1692–1715, 2010.

[26] F. Schwenker and E. Trentin, "Pattern classification and clustering: a review of partially supervised learning approaches," *Pattern Recognition Letters*, vol. 37, no. 1, pp. 4–14, 2014.

[27] M. M. Gaber and P. S. Yu, "A framework for resource-aware knowledge discovery in data streams: a holistic approach with its application to clustering," in *Proceedings of the 21st Annual ACM Aymposium on Applied Computing (SAC '06)*, pp. 649–656, ACM, Dijon, France, April 2006.

[28] P. D. Haghighi, A. Zaslavsky, S. Krishnaswamya, M. M. Gabera, and S. Lokeb, "Context-aware adaptive data stream mining," *Intelligent Data Analysis*, vol. 13, no. 3, pp. 423–434, 2009.

[29] B. Settles, "Active Learning Literature Survey, Computer Sciences," Tech. Rep. 1648, University of Wisconsin-Madison, Madison, Wis, USA, 2009.

[30] L. Findlater and J. Mcgrenere, "Impact of screen size on performance, awareness, and user satisfaction with adaptive graphical user interfaces," in *Proceedings of the CHI Conference on Human Factors in Computing Systems (CHI '08)*, pp. 1247–1256, Florence, Italy, 2008.

[31] E. Vildjiounaite, D. Schreiber, V. Kyllönen, M. Ständer, I. Niskanen, and J. Mäntyjärvi, "Prediction of interface preferences with a classifier selection approach," *Journal on Multimodal User Interfaces*, vol. 7, no. 4, pp. 321–349, 2013.

[32] U. Guz, G. Tur, D. Hakkani-Tür, and S. Cuendet, "Cascaded model adaptation for dialog act segmentation and tagging," *Computer Speech & Language*, vol. 24, no. 2, pp. 289–306, 2010.

[33] P. Dourish, "What we talk about when we talk about context," *Personal and Ubiquitous Computing*, vol. 8, no. 1, pp. 19–30, 2004.

[34] S. Tang, Y.-T. Zheng, Y. Wang, and T.-S. Chua, "Sparse ensemble learning for concept detection," *IEEE Transactions on Multimedia*, vol. 14, no. 1, pp. 43–54, 2012.

[35] L. Baltrunas and F. Ricci, "Experimental evaluation of context-dependent collaborative filtering using item splitting," *User Modeling and User-Adapted Interaction*, vol. 24, no. 1-2, pp. 7–34, 2014.

[36] B. Thomee and M. S. Lew, "Interactive search in image retrieval: a survey," *International Journal of Multimedia Information Retrieval*, vol. 1, no. 2, pp. 71–86, 2012.

[37] J. Yang, *A general framework for classifier adaptation and its applications in multimedia [Ph.D. thesis]*, 2009.

[38] T. Yao, C. W. Ngo, and S. A. Zhu, "Predicting domain adaptivity: redo or recycle?" in *Proceedings of the 20th ACM International Conference on Multimedia*, pp. 821–823, 2012.

[39] A. J. Joshi, F. Porikli, and N. P. Papanikolopoulos, "Scalable active learning for multiclass image classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, pp. 2259–2273, 2012.

[40] J. Zhang and L. Ye, "Content based image retrieval using unclean positive examples," *IEEE Transactions on Image Processing*, vol. 18, no. 10, pp. 2370–2375, 2009.

[41] J. A. Konstan and J. Riedl, "Recommender systems: from algorithms to user experience," *User Modeling and User-Adapted Interaction*, vol. 22, no. 1-2, pp. 101–123, 2012.

[42] D. H. Park, H. K. Kim, I. Y. Choi, and J. K. Kim, "A literature review and classification of recommender systems research," *Expert Systems with Applications*, vol. 39, no. 11, pp. 10059–10072, 2012.

[43] G. Adomavicius and A. Tuzhilin, "Context-aware recommender systems," in *Recommender Systems Handbook*, pp. 217–253, Springer, Berlin, Germany, 2011.

[44] M. Salamó, K. McCarthy, and B. Smyth, "Generating recommendations for consensus negotiation in group personalization services," *Personal and Ubiquitous Computing*, vol. 16, no. 5, pp. 597–610, 2012.

[45] Z. Yu, X. Zhou, Y. Hao, and J. Gu, "TV program recommendation for multiple viewers based on user profile merging," *User Modeling and User-Adapted Interaction*, vol. 16, no. 1, pp. 63–82, 2006.

[46] G. Adomavicius, R. Sankaranarayanan, S. Sen, and A. Tuzhilin, "Incorporating contextual information in recommender systems using a multidimensional approach," *ACM Transactions on Information Systems*, vol. 23, no. 1, pp. 103–145, 2005.

[47] L. Baltrunas, B. Ludwig, S. Peer, and F. Ricci, "Context relevance assessment and exploitation in mobile recommender systems," *Personal and Ubiquitous Computing*, vol. 16, no. 5, pp. 507–526, 2012.

[48] A. Jameson and B. Smyth, "Recommendation to groups," in *The Adaptive Web*, P. Brusilovsky, A. Kobsa, and W. Nejdl, Eds., vol. 4321 of *Lecture Notes in Computer Science*, pp. 596–627, Springer, Berlin, Germany, 2007.

[49] C. Senot, D. Kostadinov, M. Bouzid, J. Picault, A. Aghasaryan, and C. Bernier, "Analysis of strategies for building group profiles," in *User Modeling, Adaptation, and Personalization: 18th International Conference, UMAP 2010, Big Island, HI, USA, June 20-24, 2010. Proceedings*, vol. 6075 of *Lecture Notes in Computer Science*, pp. 40–51, Springer, Berlin, Germany, 2010.

[50] I. Garcia and L. Sebastia, "A negotiation framework for heterogeneous group recommendation," *Expert Systems with Applications*, vol. 41, no. 4, pp. 1245–1261, 2014.

[51] S. Berkovsky, T. Kuflik, and F. Ricci, "Mediation of user models for enhanced personalisation in recommender systems," *User Modeling and User-Adapted Interaction*, vol. 18, pp. 245–286, 2008.

[52] E. Vildjiounaite, V. Kyllönen, T. Hannula, and P. Alahuhta, "Unobtrusive dynamic modelling of TV programme preferences in a finnish household," *Multimedia Systems*, vol. 15, no. 3, pp. 143–157, 2009.

[53] N. Capuano, G. D'Aniello, A. Gaeta, and S. Miranda, "A personality based adaptive approach for information systems," *Computers in Human Behavior*, vol. 44, pp. 156–165, 2015.

[54] K. De Moor, T. De Pessemier, P. Mechant, C. Courtois, A. J. L. De Marez, and L. Martens, "Users' (dis)satisfaction with the personalTV application: combining objective and subjective data," *Computers in Entertainment*, vol. 9, no. 3, article 18, 2011.

[55] G. Hölbling, M. Pleschgatternig, and H. Kosch, "PersonalTV—a TV recommendation system using program metadata for content filtering," *Multimedia Tools and Applications*, vol. 46, no. 2-3, pp. 259–288, 2010.

[56] S. Stober and A. Nürnberger, "Adaptive music retrieval-a state of the art," *Multimedia Tools and Applications*, vol. 65, no. 3, pp. 467–494, 2013.

[57] J. Wagner, F. Lingenfelser, E. André, and J. Kim, "Exploring fusion methods for multimodal emotion recognition with missing data," *IEEE Transactions on Affective Computing*, vol. 2, no. 4, pp. 206–218, 2011.

[58] Z. Zeng, M. Pantic, G. I. Roisman, and T. S. Huang, "A survey of affect recognition methods: audio, visual, and spontaneous expressions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 1, pp. 39–58, 2009.

[59] B. Schuller, B. Vlasenko, F. Eyben et al., "Cross-corpus acoustic emotion recognition: variances and strategies," *IEEE Transactions on Affective Computing*, vol. 1, no. 2, pp. 119–131, 2010.

[60] M. Turk, "Multimodal interaction: a review," *Pattern Recognition Letters*, vol. 36, no. 1, pp. 189–195, 2014.

[61] S. Ronkainen, E. Koskinen, Y. Liu, and P. Korhonen, "Environment analysis as a basis for designing multimodal and multidevice user interfaces," *Human—Computer Interaction*, vol. 25, no. 2, pp. 148–193, 2010.

[62] D. Anand and K. K. Bharadwaj, "Adaptive user similarity measures for recommender systems: a genetic programming approach," in *Proceedings of the 3rd IEEE International Conference on Computer Science and Information Technology (ICCSIT '10)*, pp. 121–125, July 2010.

[63] H. Luan, Y.-T. Zheng, M. Wang, and T.-S. Chua, "VisionGo: towards video retrieval with joint exploration of human and computer," *Information Sciences*, vol. 181, no. 19, pp. 4197–4213, 2011.

[64] L. Kuncheva, *Combining Pattern Classifiers, Methods and Algorithms*, Wiley, Hoboken, NJ, USA, 2004.

[65] F. D. Macías-Escrivá, R. Haber, R. del Toro, and V. Hernandez, "Self-adaptive systems: a survey of current approaches, research challenges and applications," *Expert Systems with Applications*, vol. 40, no. 18, pp. 7267–7279, 2013.

[66] E. Morvant, A. Habrard, and S. Ayache, "Parsimonious unsupervised and semi-supervised domain adaptation with good similarity functions," *Knowledge and Information Systems*, vol. 33, no. 2, pp. 309–349, 2012.

[67] R. T. Calumby, R. da Silva Torres, and M. A. Gonçalves, "Multimodal retrieval with relevance feedback based on genetic programming," *Multimedia Tools and Applications*, vol. 69, no. 3, pp. 991–1019, 2014.

[68] P.-Y. Yin, B. Bhanu, K.-C. Chang, and A. Dong, "Integrating relevance feedback techniques for image retrieval using reinforcement learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1536–1551, 2005.

[69] K. Z. Gajos, D. S. Weld, and J. O. Wobbrock, "Automatically generating personalized user interfaces with Supple," *Artificial Intelligence*, vol. 174, no. 12-13, pp. 910–950, 2010.

[70] J. Kong, W. Y. Zhang, N. Yu, and X. J. Xia, "Design of human-centric adaptive multimodal interfaces," *International Journal of Human Computer Studies*, vol. 69, no. 12, pp. 854–869, 2011.

[71] M. Macik, T. Cerny, and P. Slavik, "Context-sensitive, cross-platform user interface generation," *Journal on Multimodal User Interfaces*, vol. 8, no. 2, pp. 217–229, 2014.

[72] K. Z. Gajos, J. O. Wobbrock, and D. S. Weld, "Improving the performance of motor-impaired users with automatically-generated, ability-based interfaces," in *Proceedings of the 26th Annual SIGCHI Conference on Human Factors in Computing Systems (CHI '08)*, pp. 1257–1266, ACM, April 2008.

[73] M. Ferecatu, N. Boujemaa, and M. Crucianu, "Semantic interactive image retrieval combining visual and conceptual content description," *Multimedia Systems*, vol. 13, no. 5-6, pp. 309–322, 2008.

[74] Y. Bengio, "Learning deep architectures for AI," *Foundations and Trends in Machine Learning*, vol. 2, no. 1, pp. 1–27, 2009.

[75] E. Vildjiounaite, V. Kyllönen, S.-M. Mäkelä et al., "Semi-supervised context adaptation: case study of audience excitement recognition," *Multimedia Systems*, vol. 18, no. 3, pp. 231–250, 2012.

[76] Y.-G. Jiang, Q. Dai, J. Wang, C.-W. Ngo, X. Xue, and S.-F. Chang, "Fast semantic diffusion for large-scale context-based image and video annotation," *IEEE Transactions on Image Processing*, vol. 21, no. 6, pp. 3080–3091, 2012.

[77] O. Pauplin, P. Caleb-Solly, and J. Smith, "User-centric image segmentation using an interactive parameter adaptation tool," *Pattern Recognition*, vol. 43, no. 2, pp. 519–529, 2010.

[78] G. Caridakis, K. Karpouzis, and S. Kollias, "User and context adaptive neural networks for emotion recognition," *Neurocomputing*, vol. 71, no. 13–15, pp. 2553–2562, 2008.

[79] http://emotion-research.net/toolbox/toolboxdatabase.2006-09-26.5667892524.

[80] W. Jiang, E. Zavesky, Sh.-F. Chang, and A. Loui, "Cross-domain learning methods for high-level visual concept classification," in *Proceedings of the IEEE International Conference on Image Processing (ICIP '08)*, pp. 161–164, IEEE, San Diego, Calif, USA, October 2008.

[81] J. Yang, R. Yan, and A. G. Hauptmann, "Cross-domain video concept detection using adaptive SVMs," in *Proceedings of the*

15th ACM International Conference on Multimedia (MM '07), pp. 188–197, September 2007.

[82] D. Zhang, D. Gatica-Perez, S. Bengio, and I. McCowan, "Semi-supervised meeting event recognition with adapted HMMs," in *Proceedings of the IEEE Conference on Multimedia and Expo (ICME '05)*, pp. 611–618, July 2005.

[83] Y. Blanco-Fernández, M. López-Nores, J. Pazos-Arias, A. Gil-Solla, and M. Ramos-Cabrer, "Exploiting digital TV users' preferences in a tourism recommender system based on semantic reasoning," *IEEE Transactions on Consumer Electronics*, vol. 56, no. 2, pp. 904–912, 2010.

[84] E. Vildjiounaite, V. Kyllönen, and J. Mäntyjärvi, "If their car talks to them, shall a kitchen talk too? Cross-context mediation of interaction preferences," in *Proceedings of the 3rd ACM SIGCHI Symposium on Engineering Interactive Computing Systems (EICS '11)*, pp. 111–116, ACM, June 2011.

[85] N. Hariri, B. Mobasher, and R. Burke, "Context-aware music recommendation based on latent topic sequential patterns," in *Proceedings of the 6th ACM Conference on Recommender Systems (RecSys '12)*, pp. 131–138, September 2012.

[86] U. Panniello, A. Tuzhilin, and M. Gorgoglione, "Comparing context-aware recommender systems in terms of accuracy and diversity," *User Modeling and User-Adapted Interaction*, vol. 24, no. 1-2, pp. 35–65, 2014.

[87] Y.-L. Chen, L.-C. Cheng, and C.-N. Chuang, "A group recommendation system with consideration of interactions among group members," *Expert Systems with Applications*, vol. 34, no. 3, pp. 2082–2090, 2008.

[88] G. S. Thyagaraju and U. P. Kulkarni, "Family aware TV program and settings recommender," *International Journal of Computer Applications*, vol. 29, no. 4, pp. 1–18, 2011.

[89] E. Apeh and B. Gabrys, "Detecting and visualizing the change in classification of customer profiles based on transactional data," *Evolving Systems*, vol. 4, no. 1, pp. 27–42, 2013.

[90] I. T. Christou, G. Gekas, and A. Kyrikou, "A classifier ensemble approach to the TV-viewer profile adaptation problem," *International Journal of Machine Learning and Cybernetics*, vol. 3, no. 4, pp. 313–326, 2012.

[91] H.-C. Shih, J.-N. Hwang, and C.-L. Huang, "Content-based attention ranking using visual and contextual attention model for baseball videos," *IEEE Transactions on Multimedia*, vol. 11, no. 2, pp. 244–255, 2009.

[92] J. Chen, X. Liu, P. Tu, and A. Aragones, "Learning person-specific models for facial expression and action unit recognition," *Pattern Recognition Letters*, vol. 34, no. 15, pp. 1964–1970, 2013.

[93] P.-Y. Yin, "Particle swarm optimization for automatic selection of relevance feedback heuristics," in *Advances in Swarm Intelligence: First International Conference, ICSI 2010, Beijing, China, June 12–15, 2010, Proceedings, Part I*, vol. 6145 of *Lecture Notes in Computer Science*, pp. 167–174, Springer, Berlin, Germany, 2010.

[94] H. Li, Y. Shi, Y. Liu, A. G. Hauptmann, and Z. Xiong, "Cross-domain video concept detection: a joint discriminative and generative active learning approach," *Expert Systems with Applications*, vol. 39, no. 15, pp. 12220–12228, 2012.

[95] A. S. Britto Jr., R. Sabourin, and L. E. S. de Oliveira, "Dynamic selection of classifiers—a comprehensive review," *Pattern Recognition*, vol. 47, no. 11, pp. 3665–3680, 2014.

[96] I. Mporas, T. Ganchev, O. Kocsis, and N. Fakotakis, "Context-adaptive pre-processing scheme for robust speech recognition in fast-varying noise environment," *Signal Processing*, vol. 91, no. 8, pp. 2101–2111, 2011.

[97] I. Žliobaite, J. Bakker, and M. Pechenizkiy, "Beating the baseline prediction in food sales: how intelligent an intelligent predictor is?" *Expert Systems with Applications*, vol. 39, no. 1, pp. 806–815, 2012.

[98] P. R. Cavalin, R. Sabourin, and C. Y. Suen, "LoGID: an adaptive framework combining local and global incremental learning for dynamic selection of ensembles of HMMs," *Pattern Recognition*, vol. 45, no. 9, pp. 3544–3556, 2012.

[99] X. Li and Q. Ji, "Active affective state detection and user assistance with dynamic Bayesian networks," *IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans.*, vol. 35, no. 1, pp. 93–105, 2005.

[100] Y. Wu, E. Y. Chang, and B. L. Tseng, "Multimodal metadata fusion using causal strength," in *Proceedings of the 13th ACM International Conference on Multimedia*, pp. 872–881, Singapore, November 2005.

[101] A. Gallagher and T. Chen, "Using context to recognize people in consumer images," *IPSJ Journal*, vol. 49, pp. 1234–1245, 2008.

[102] C. Palmisano, A. Tuzhilin, and M. Gorgoglione, "Using context to improve predictive modeling of customers in personalization applications," *IEEE Transactions on Knowledge and Data Engineering*, vol. 20, no. 11, pp. 1535–1549, 2008.

[103] G.-J. Qi, J. Tang, M. Wang et al., "Correlative multilabel video annotation with temporal kernels," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 5, no. 1, article 3, 27 pages, 2008.

[104] G. Papadopoulos, V. Mezaris, I. Kompatsiaris, and M. G. Strintzis, "Joint modality fusion and temporal context exploitation for semantic video analysis," *EURASIP Journal on Advances in Signal Processing*, vol. 2011, article 89, 2011.

[105] M.-F. Weng and Y.-Y. Chuang, "Cross-domain multicue fusion for concept-based video indexing," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 10, pp. 1927–1941, 2012.

[106] A. Metallinou, M. Wöllmer, A. Katsamanis, F. Eyben, B. Schuller, and S. Narayanan, "Context-sensitive learning for enhanced audiovisual emotion classification," *IEEE Transactions on Affective Computing*, vol. 3, no. 2, pp. 184–198, 2012.

[107] M. Wöllmer, A. Metallinou, F. Eyben, B. Schuller, and S. Narayanan, "Context-sensitive multimodal emotion recognition from speech and facial expression using bidirectional LSTM modeling," in *Proceedings of the 11th Annual Conference of the International Speech Communication Association: Spoken Language Processing for All (INTERSPEECH '10)*, pp. 2362–2365, September 2010.

[108] B. Schuller, R. Müller, F. Eyben et al., "Being bored? Recognising natural interest by extensive audiovisual integration for real-life application," *Image and Vision Computing*, vol. 27, no. 12, pp. 1760–1774, 2009.

[109] K. Forbes-Riley and D. Litman, "Predicting emotion in spoken dialogue from multiple knowledge sources," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL '04)*, pp. 201–208, 2004.

[110] R. López-Cózar, J. Silovsky, and M. Kroul, "Enhancement of emotion detection in spoken dialogue systems by combining several information sources," *Speech Communication*, vol. 53, no. 9-10, pp. 1210–1228, 2011.

[111] D. Griol, J. M. Molina, and Z. Callejas, "Modeling the user state for context-aware spoken interaction in ambient assisted living," *Applied Intelligence*, vol. 40, no. 4, pp. 749–771, 2014.

[112] F. S. da Silva, L. G. P. Alves, and G. Bressan, "Personal TVware: an infrastructure to support the context-aware recommendation for personalized digital TV," *International Journal of Computer Theory and Engineering*, vol. 4, no. 2, pp. 131–136, 2012.

[113] E. Vildjiounaite, V. Kyllönen, T. Hannula, and P. Alahuhta, "Unobtrusive dynamic modelling of TV program preferences in a household," in *Proceedings of the 6th European Conference on Changing Television Environments (EUROITV '08)*, pp. 82–91, Salzburg, Austria, July 2008.

[114] H. Ahn, K.-J. Kim, and I. Han, "Mobile advertisement recommender system using collaborative filtering: MAR-CF," Technical Report of the Korea Society of Management Information Systems, The Korea Society of Management Information Systems, 2006.

[115] R. Rafeh and A. Bahrehmand, "An adaptive approach to dealing with unstable behaviour of users in collaborative filtering systems," *Journal of Information Science*, vol. 38, no. 3, pp. 205–221, 2012.

[116] S. Kirstein, H. Wersing, H.-M. Gross, and E. Körner, "A life-long learning vector quantization approach for interactive learning of multiple categories," *Neural Networks*, vol. 28, pp. 90–105, 2012.

[117] A. Jain and A. Kumar, "Biometric recognition: an overview," in *Second Generation Biometrics: The Ethical, Legal and Social Context, the International Library of Ethics, Law and Technology*, vol. 11, pp. 49–79, Springer, 2012.

[118] F. Deravi, "Intelligent biometrics," in *Second Generation Biometrics: The Ethical, Legal and Social Context*, vol. 11 of *The International Library of Ethics, Law and Technology*, pp. 177–191, Springer, Dordrecht, The Netherlands, 2012.

[119] O. Fatukasi, J. Kittler, and N. Poh, "Estimation of missing values in multimodal biometric fusion," in *Proceedings of the IEEE 2nd International Conference on Biometrics: Theory, Applications and Systems (BTAS '08)*, pp. 1–6, October 2008.

[120] R. Vera-Rodriguez, P. Tome, J. Fierrez, and J. Ortega-Garcia, "Fusion of footsteps and face biometrics on an unsupervised and uncontrolled environment," in *Sensing Technologies for Global Health, Military Medicine, Disaster Response, and Environmental Monitoring II; and Biometric Technology for Human Identification IX*, vol. 8371 of *Proceedings of SPIE*, Baltimore, Md, USA, April 2012.

[121] M. Aste, M. Boninsegna, A. Freno, and E. Trentin, "Techniques for dealing with incomplete data: a tutorial and survey," *Pattern Analysis and Applications*, vol. 18, no. 1, pp. 1–29, 2015.

[122] J. Wang, Y. Li, and C. Wang, "How to handle missing data in robust multi-biometrics verification," *International Journal of Biometrics*, vol. 3, no. 3, pp. 265–283, 2011.

[123] N. Poh and J. Kittler, "A Unified framework for biometric expert fusion incorporating quality measures," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 1, pp. 3–18, 2012.

Series title and number

VTT Science 125

Title	**Lightweight adaptation to situational changes in classifiers of multimodal human data**
Author(s)	Elena Vildjiounaite
Abstract	A wide range of current computer applications require explicit human-computer interaction of various types, ranging from application login to providing feedback on the applications' outputs (e.g., ranking recommended TV programmes), to adapt to different usage situations. As explicit interaction can be tiresome, users tend to avoid it even if such behaviour is counterproductive and/or insecure. Accordingly, application developers rarely build systems capable of runtime adaptation to new contexts, because conventional classifier training requires too large datasets of labelled training data to obtain from end users. The most common adaptation schemes define one or more typical usage contexts, build a pool of context-specific reasoning models during the design time and then select an appropriate model from this pool during the runtime. This approach enables instant runtime adaptation, but requires domain knowledge and suits only applications with usage contexts that can be pre-defined. Many personal applications, however, encounter wide varieties of difficult-to-define contexts, e.g. social rules or audio backgrounds. It is simply impossible to predict all such contexts, to say nothing of collecting adequate databases for building pools of reasoning models for them. Hence personal applications require new methods for adapting to changing runtime contexts. As runtime adaptation largely relies on interaction with end users, these methods should be fairly lightweight with respect to standard ones, i.e. they should require much less domain knowledge and explicitly acquired data. This thesis introduces and explores lightweight solutions for building reasoning models for situations that are not pre-defined during the design time. These solutions are proposed for increasing the accuracy or convenience of applications in three domains: TV programme recommendations, affect recognition and personal assistance. In addition, a method for reducing explicit interaction efforts at the inference stage is proposed for increasing the security of biometric verification systems in which typical usage contexts can be pre-defined. The proposed methods have been validated experimentally with realistic data sets, and the results have confirmed that they considerably reduce the dependence of context- and user-adaptive classifiers on domain knowledge and explicit interaction efforts. Studies with personal assistive applications have also demonstrated that users can accept the proposed lightweight adaptation even when its accuracy is relatively low. The diversity of test cases, which differed considerably in their requirements and data availability, made it possible to demonstrate how the suitability of different adaptation schemes depends on both the application and its usage contexts. Based on this experience, this thesis identifies context and application characteristics that exercise the greatest influence and provides guidelines for considering these factors in adaptation design.
ISBN, ISSN, URN	ISBN 978-951-38-8410-9 (Soft back ed.) ISBN 978-951-38-8411-6 (URL: http://www.vttresearch.com/impact/publications) ISSN-L 2242-119X ISSN 2242-119X (Print) ISSN 2242-1203 (Online) http://urn.fi/URN:ISBN:978-951-38-8411-6
Date	May 2016
Language	English, Finnish abstract
Pages	89 p. + app. 120 p.
Name of the project	
Commissioned by	
Keywords	human-computer interaction, context adaptation, multimodal fusion
Publisher	VTT Technical Research Centre of Finland Ltd P.O. Box 1000, FI-02044 VTT, Finland, Tel. 020 722 111

Julkaisun sarja ja numero

VTT Science 125

Nimeke	**Multimodaalisen käyttäjädatan luokittelijan kevyt mukauttaminen muuttuvissa tilanteissa**
Tekijä(t)	Elena Vildjiounaite
Tiivistelmä	Monet nykyisistä tietokonesovelluksista edellyttävät erilaista eksplisiittistä interaktiota ihmisen ja tietokoneen välillä. Tämä pitää sisällään esimerkiksi sovelluksiin kirjautumisen ja sovellusten toimintaa koskevan palautteen antamisen (esim. suositeltujen televisio-ohjelmien asettamisen paremmuusjärjestykseen) sovellusten mukauttamiseksi. Koska eksplisiittinen interaktio voi olla rasittavaa, käyttäjät mielellään välttävät sitä, vaikka tämä olisikin haitallista ja/tai tietoturvaa heikentävää. Sovellusten kehittäjät rakentavat sen vuoksi harvoin järjestelmiä, jotka pystyvät mukautumaan uusiin konteksteihin käytönaikaisesti, koska perinteinen luokittajan ohjattu opettaminen edellyttää liian suurien annotoitujen tietojoukkojen hankkimista loppukäyttäjiltä. Tavallisimmissa mukautumismalleissa määritetään yksi tai useampi tyypillinen käyttökonteksti ja rakennetaan suunnitteluaikana kontekstikohtaisten päättelymallien varanto, josta valitaan sopiva malli käytön aikana. Tämä lähestymistapa mahdollistaa välittömän käytönaikaisen mukautumisen, mutta edellyttää tietämystä toimialueista ja soveltuu ainoastaan sovelluksille, joissa käyttökontekstit on mahdollista määrittää etukäteen. Useita henkilökohtaisia sovelluksia käytettäessä vastaan tulee kuitenkin monia erilaisia, vaikeasti määritettäviä konteksteja, kuten sosiaalisia sääntöjä tai äänitaustoja. Kaikkia tällaisia konteksteja ei yksinkertaisesti ole mahdollista ennustaa, puhumattakaan niiden vaatimien päättelymallien varantoon riittävien tietokantojen keräämisestä. Tästä syystä henkilökohtaisia sovelluksia varten tarvitaan uusia menetelmiä, jotka mahdollistavat mukautumisen käytön aikana muuttuviin konteksteihin. Koska käytönaikainen mukautuminen pohjautuu pääasiallisesti sovellusten ja loppukäyttäjien interaktioon, näiden menetelmien on oltava melko kevyitä standardinmukaisiin menetelmiin nähden. Toisin sanoen niiden on edellytettävä vähemmän tietämystä toimialueista ja vähemmän eksplisiittisesti kerättyjä tietoja. Tässä tutkielmassa esitellään ja tutkitaan kevyitä ratkaisuja päättelymallien rakentamiseen sellaisia tilanteita varten, joita ei ole määritetty ennalta suunnitteluaikana. Näitä ratkaisuja ehdotetaan sovellusten tarkkuuden tai kätevyyden parantamiseksi kolmella toimialueella: televisio-ohjelmia koskevat suositukset, tunteiden tunnistaminen ja henkilökohtaiset avustustoiminnot. Lisäksi ehdotetaan menetelmää, jonka avulla voidaan vähentää eksplisiittistä interaktiota päättelyvaiheessa suojauksen parantamiseksi biometrisissä todentamisjärjestelmissä, joissa tyypilliset käyttökontekstit ovat etukäteen määritettävissä. Ehdotetut menetelmät on vahvistettu kokeellisesti realististen aineistojen avulla. Saadut tulokset vahvistavat, että menetelmien avulla on pystytty tuntuvasti vähentämään kontekstin ja käyttäjän mukaan mukautuvien luokittajien riippuvuutta toimialuetietämyksestä ja eksplisiittisestä interaktiosta. Henkilökohtaisia avustavia sovelluksia koskevissa tutkimuksissa on myös osoitettu, että käyttäjät hyväksyvät ehdotetun kevyen mukautuksen, vaikka sen tarkkuus olisi suhteellisen heikko. Koska testitapaukset olivat niin monimuotoisia ja poikkesivat huomattavasti toisistaan vaatimusten ja käytettävissä olevien tietojen osalta, oli mahdollista osoittaa, miten riippuvaista erilaisten mukautumismallien soveltuvuus on sekä itse sovelluksesta että sen käyttökonteksteista. Näiden kokemusten pohjalta tutkielmassa tunnistetaan kontekstien ja sovellusten ominaisuuksia, joilla on suurin vaikutus, sekä tarjotaan suosituksia siitä, miten nämä tekijät voidaan huomioida mukautumissuunnittelussa.
ISBN, ISSN, URN	ISBN 978-951-38-8410-9 (nid.) ISBN 978-951-38-8411-6 (URL: http://www.vtt.fi/julkaisut) ISSN-L 2242-119X ISSN 2242-119X (Painettu) ISSN 2242-1203 (Verkkojulkaisu) http://urn.fi/URN:ISBN:978-951-38-8411-6
Julkaisuaika	Toukokuu 2016
Kieli	Englanti, suomenkielinen tiivistelmä
Sivumäärä	89 s. + liitt. 120 s.
Projektin nimi	
Rahoittajat	
Avainsanat	human-computer interaction, context adaptation, multimodal fusion
Julkaisija	Teknologian tutkimuskeskus VTT Oy PL 1000, 02044 VTT, puh. 020 722 111

Lightweight adaptation to situational changes in classifiers of multimodal human data

Intelligent computer applications need to adapt their behaviour to contexts and users, but conventional methods to train multimodal classifiers do not suit to this purpose because they require acquiring large sets of labelled data for each situation. Due to large variety of usage contexts of personal applications, no developer can predict all these situations, to say nothing of collecting adequate training databases for them. Hence personal applications require new methods for adapting to changing runtime contexts. As runtime adaptation largely relies on interaction with end users, these methods should be fairly lightweight with respect to standard ones, i.e. they should require much less domain knowledge and explicitly acquired data.

This thesis introduces lightweight solutions for adapting reasoning models to situations at runtime, identifies important context and application characteristics and provides guidelines for considering these factors in adaptation design. The proposed solutions have been validated experimentally with realistic data sets, and the results have confirmed that they considerably reduce the dependence of context- and user-adaptive classifiers on domain knowledge and explicit interaction efforts. Studies with personal assistive applications have also demonstrated that users can accept the proposed lightweight adaptation even when its accuracy is relatively low.

ISBN 978-951-38-8410-9 (Soft back ed.)
ISBN 978-951-38-8411-6 (URL: http://www.vttresearch.com/impact/publications)
ISSN-L 2242-119X
ISSN 2242-119X (Print)
ISSN 2242-1203 (Online)
http://urn.fi/URN:ISBN:978-951-38-8411-6

VTT SCIENCE 125

Lightweight adaptation to situational changes in...