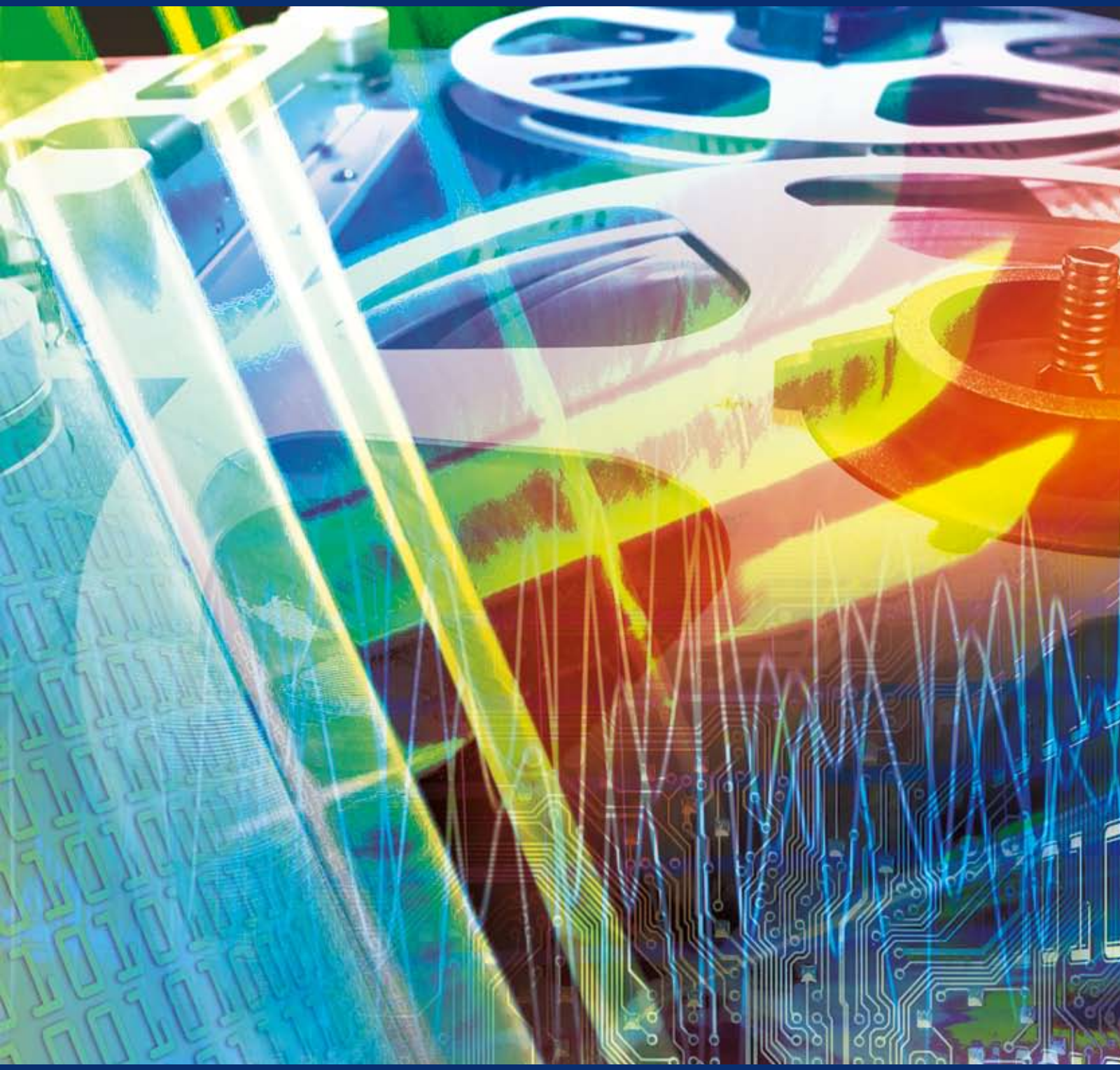
EURASIP Journal on Image and Video Processing

Facial Image Processing

Guest Editors: Christophe Garcia, Jörn Ostermann, and Tim Cootes



Facial Image Processing

EURASIP Journal on Image and Video Processing

Facial Image Processing

Guest Editors: Christophe Garcia,
Jörn Ostermann, and Tim Cootes

Contents

Hindawi Publishing Corporation
EURASIP Journal on Image and Video Processing
Volume 2007, Article ID 70872, 2 pages
doi:10.1155/2007/70872

Editorial
Facial Image Processing

Christophe Garcia,[1] Jörn Ostermann,[2] and Tim Cootes[3]

[1] *Laboratory of Image, Rich Media and Hyperlanguages, Orange Labs, France Telecom R&D, 35510 Cession-Sévigné, Rennes, France*
[2] *Institut für Informationsverarbeitung, Leibniz Universität Hannover, 30167 Hannover, Germany*
[3] *Division of Imaging Science and Biomedical Engineering, University of Manchester, Manchester M13 9PL, UK*

Received 12 December 2007; Accepted 12 December 2007

Facial image processing is an area of research dedicated to the extraction and analysis of information about human faces; information which is known to play a central role in social interactions including recognition, emotion, and intention.

Over the last decade, it has become a very active research field that deals with face detection and tracking, facial feature detection, face recognition, facial expression and emotion recognition, face coding, and virtual face synthesis.

With the introduction of new powerful machine learning techniques, statistical classification methods, and complex deformable models, recent progresses have made possible a large number of applications in areas such as image retrieval, surveillance and biometrics, visual speech understanding, virtual characters for e-learning, online marketing or entertainment, intelligent human-computer interaction, and others.

However, much remains to be done to provide more robust systems, especially when dealing with pose and illumination changes in complex natural scenes. If most approaches focus naturally on processing from still images, emerging techniques may also consider different inputs. For instance, video is becoming ubiquitous and very affordable, and there is a growing demand for vision-based human oriented applications, ranging from security to human computer interaction and video annotation. Capturing 3D data may as well become very affordable and processing such data can lead to enhanced systems, more robust to illumination effects and where discriminant information may be more easily retrieved.

The scope of this special issue of the EURASIP Journal on Image and Video Processing is to present original contributions in the field of facial image processing, and especially on face verification and recognition, facial feature detection, face synthesis, and 3D face acquisition.

Among the 20 submitted papers, six articles have been selected for this special issue.

The paper by Arya and DiPaola addresses the construction of a behavioral face model for affective social agents based on three independent but interacting parameter spaces which are knowledge, personality, and mood. While a geometry space provides an MPEG-4 compatible set of parameters for low-level control, the behavioral extensions available through the triple spaces provide flexible means of designing complicated personality types, facial expression, and dynamic interactive scenarios.

Robust facial feature detection for facial expression recognition in uncontrolled environments is the focus of investigation in the work presented by Ioannou et al. The proposed system is based on a multicue feature extraction and fusion technique, which provides MPEG-4-compatible features assorted with a confidence measure, used to weight their importance in the recognition of the observed facial expression, while the fusion process ensures that the final result will be based on the extraction technique that performed better given the particular lighting or color conditions.

Mitéran et al. address 3D face acquisition, which is becoming of great importance in face recognition, virtual reality, and many other applications. They propose a new real-time stereo vision system that provides a dense face disparity map, based on a hybrid architecture (FPGA-DSP) allowing a real-time embedded and reconfigurable processing.

The paper by Wang et al. focuses on the fusion of 2D facial images and 3D stereo depth maps for enhancing face recognition. They propose an original machine learning method, the bilateral two-dimensional linear discriminant analysis (B2DLDA), able to extract discriminant facial features from the appearance and disparity images. They show that present-day passive stereoscopy does make a positive contribution to face recognition.

Ciocoiu and Costin study different localized representation and manifold learning approaches for face recognition. They conduct a systematic comparative analysis in terms of distance metrics, number of selected features, and sources of variability on the AR and Olivetti face databases. The reported results indicate that the relative ranking of the methods is highly task dependent, and the performances vary significantly according to the selected distance metric.

Finally, Lee and Sohn tackle the problem of multiview face recognition. Many current face descriptors give satisfactory results with frontal views, but fail to accurately represent all views of the human head. The authors propose a new paradigm to facilitate multiview face recognition, not through a multiview face recognizer, but through multiple single-view recognizers. The resulting face descriptor based on multiple representative views, which is of compact size, provides reasonable face recognition performance on any facial view.

To conclude, we would like to thank the authors, reviewers, and the editorial team of the EURASIP Journal on Image and Video Processing for their effort in the preparation of this special issue. We hope this issue allows the reader to get an insight in the recent advances on facial image processing and stimulates the cross-fertilization that has been ongoing between the image analysis and image synthesis communities.

Christophe Garcia
Jörn Ostermann
Tim Cootes

Hindawi Publishing Corporation
EURASIP Journal on Image and Video Processing
Volume 2007, Article ID 48757, 12 pages
doi:10.1155/2007/48757

Research Article

Multispace Behavioral Model for Face-Based Affective Social Agents

Ali Arya[1] and Steve DiPaola[2]

[1] *Carleton School of Information Technology, Carleton University, Ottawa, ON, Canada K1S5B6*
[2] *School of Interactive Arts & Technology, Simon Fraser University, Surrey, BC, Canada V3TOA3*

Received 26 April 2006; Revised 4 October 2006; Accepted 22 December 2006

Recommended by Tim Cootes

This paper describes a behavioral model for affective social agents based on three independent but interacting parameter spaces: knowledge, personality, and mood. These spaces control a lower-level geometry space that provides parameters at the facial feature level. Personality and mood use findings in behavioral psychology to relate the perception of personality types and emotional states to the facial actions and expressions through two-dimensional models for personality and emotion. Knowledge encapsulates the tasks to be performed and the decision-making process using a specially designed XML-based language. While the geometry space provides an MPEG-4 compatible set of parameters for low-level control, the behavioral extensions available through the triple spaces provide flexible means of designing complicated personality types, facial expression, and dynamic interactive scenarios.

1. INTRODUCTION

Chuck Jones, the cocreator of such legendary animated characters as Bugs Bunny, Daffy Duck, and the Road Runner, once said [22]: "Believability. That is what we were striving for." The history of animation, traditional or computer-generated, has shown that the most successful animated characters are not necessarily those who have been geometrically realistic, but those that are believable in behavior. As many researchers in the area of social agents have noticed [4, 5, 29], this believability of characters (i.e., acting in a realistic and "natural" way) is a key element in allowing viewers/users to relate to the agent. In our opinion, such believability depends, mainly, on proper behavioral modeling. Another aspect of behavioral modeling is the creation of nonscripted actions. A strong behavioral model allows an animated character such as a social agent to follow certain rules or high-level scripts, and define and create proper details of actions based on any dynamic situation with no need to design those details in advance.

Although many researchers have proposed behavioral models for social agents [4, 5, 17, 19, 26, 33, 36], the following essential features seem to require further improvements.

(1) Theoretical base in behavioral psychology.
(2) Proper parameterization to simplify the model configuration.
(3) Scripting language specially designed for agents.
(4) Independence of behavioral components such as tasks, personality, and mood.

In this paper, we describe the behavioral model used in our facial animation system, iFACE (interactive face animation—comprehensive environment) [2]. iFACE uses a parameterized approach where the behavior is controlled through three separate but interacting parameter spaces: knowledge, personality, and mood (see Figure 1). They are not organized as layers on top of each other; they are "parallel" which means that each one can operate (and be controlled) independently while at same time interact with the other ones. Knowledge is the primary space where all action and configuration scripts are processed. Personality and mood can be controlled by these scripts and personality itself can affect mood. A fourth parameter space, geometry, forms the visual foundation of the system with low-level parameters such as size and location of facial features. A hierarchical set of geometrical parameters provides an efficient and unified set of controls for facial actions,

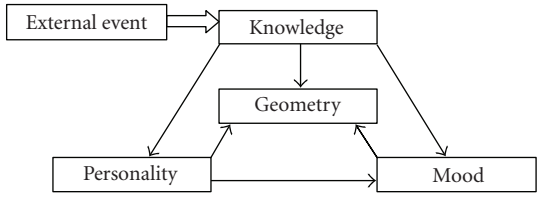


Figure 1: behavioral model parameter spaces.

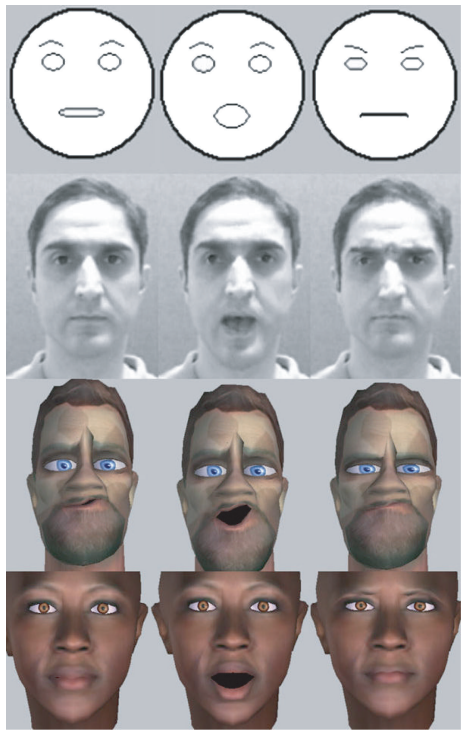


Figure 2: Sample animated heads from iFACE, featuring neutral, talking, and frowning states (columns 1 to 3, resp.) of 2D cartoonish, 2D photorealistic, 3D cartoonish, and 3D realistic faces (rows 1 to 4, resp.).

independent of the 2D or 3D head data type, as shown in Figure 2.

Knowledge encapsulates the tasks to be performed and general rules of behavior that are independent of the character. A specially designed XML-based language is used for knowledge space. Personality and mood are based on parameterized models in behavioral psychology and represent the characteristics and emotional state of a specific individual. Personality is related to the long-term traits such as typical head movements and mood controls short-term emotional states visualized by facial expressions.

The principal concept in our research is that parameterization allows animators and designers to create new geometries, personality types, and emotional states without being involved in technical details. For example, changing the affiliation and dominance [40] parameters can easily create new personalities, and since the parameters are associated to facial actions, this new personality type already has proper facial actions. The existing systems either do not use well-defined and scientifically accepted parameters or have not associated the parameters to facial actions properly (e.g., random or ad-hoc selection of actions compared to our system that is based on user studies with the aid of behavioral psychologists). So our main contributions, compared to the existing research that we will review later, are the following.

(1) The only XML-based face-specific language compatible with MPEG-4 with dynamic decision-making and temporal constructs.
(2) Associating facial actions to the perceived personality based on user studies. Facial actions have been extensively studied with regards to emotions but not personality.
(3) Linking facial actions to personality and emotion parameters rather than "personality types" and "emotional states" themselves. As we will see, this will cause facial actions that are more "perceptually valid" when creating new and combined types and states.
(4) A layered geometry model that allows animation parameters and design files to be applied to a variety of data types (see Figure 2) due to abstraction and hiding details.
(5) A unified model encapsulating all required features in one framework.

In Section 2, we review some of the related research in the area of behavioral modeling for social agents. Sections 3 to 7 discuss our proposed behavioral model in detail. Two example applications of iFACE system and its behavioral model are the subject of Section 8, and some concluding remarks are presented in Section 9.

2. RELATED WORK

2.1. Agent and multimedia languages

The facial action coding system (FACS) was the earliest approach to systematically describe facial action in terms of small action units (AUs) such as left-eye-lid-close and jaw-open [19]. The MPEG-4 standard [6] extended this idea and introduced face definition parameters (FDPs) and face animation parameters (FAPs). FDPs define a face by giving measures for its major parts and features such as eyes, lips, and their related distances. FAPs on the other hand, encode the movements of these facial features. Together they allow a receiver system to create a face (using any graphics method) and animate that face based on low-level commands in FAPs.

Synchronized multimedia integration language (SMIL) [12] is an XML-based language designed to specify temporal relationships of components in a multimedia presentation, especially in web applications. SMIL can coexist quite suitably with MPEG-4 object-based streams. SMIL animation is a newer language (http://www.w3.org/TR/smil-animation) based on SMIL, which is aimed at describing animation pieces. It establishes a framework for general animation but neither of these two provides any specific means for facial

```
<vhml>
<person disposition=‘‘angry’’>
<p>
First I speak with an angry voice,
<surprised intensity=‘‘50’’>
then I change to look surprised.
</surprised>
</p></person>
</vhml>
```

Figure 3: An example of VHML script.

animation. There have also been different languages in the fields of virtual reality and computer graphics for modeling computer-generated scenes. Examples are virtual reality modeling language (VRML, http://www.vrml.org) and programming libraries like OpenGL (http://www.opengl.org).

These languages are not customized for face animation, and do not provide any explicit support for it. The absence of a dedicated language for face animation, as an abstraction on top of FACS AUs or MPEG-4 FAPs has drawn attention to the development of markup languages for virtual characters [1, 15, 30, 35]. Virtual human markup language (VHML) [30] is an XML-based language for the representation of different aspects of "virtual humans," that is, avatars, such as speech production, facial and body animation, emotional representation, dialogue management, and hyper and multimedia information (http://www.vhml.org). It comprises a number of special-purpose languages for emotion and facial and body animation. In VHML, timing of animation elements in relation to each other and in relation to the realization of text is achieved via the attributes "`duration`" and "`wait`." A simple VHML document is shown in Figure 3.

Multimodal presentation markup language (MPML) [35] is another XML-based markup language developed to enable the description of multimodal presentation in a web browser, based on animated characters (http://www.miv.t.u-tokyo.ac.jp/MPML/en). It offers functionalities for synchronizing media presentation (reusing parts of the synchronized multimedia integration language, SMIL) and new XML elements such as `<listen>` (basic interactivity), `<test>` (decision making), `<speak>` (spoken by a TTS system), `<move>` (to a certain point at the screen), and `<emotion>` (for standard facial expressions). MPML addresses the interactivity and decision making not directly covered by VHML/FAML, but both suffer from a lack of explicit compatibility with MPEG-4 (XMT, FAPs, etc.).

2.2. Personality and perception

behavioral psychologists have studied human personality and its models and parameters for quite a while. Many personality models have been proposed, and one of the most notable examples is the big five or five-factor model [21, 39]. The big-5 model considers five major personality dimensions: openness, conscientiousness, extraversion, agreeableness, and neuroticism (OCEAN). Modeling personality as an N-dimensional space allows for navigating through the personality space by changing one parameter along each independent dimension. Although successful in many aspects, the five dimensions in the Big-5 model are (1) not independent enough and (2) hard to visualize. This results in the model being hard to use for animated characters needing user-friendly and controllable personality parameters. Wiggins et al. [40] have proposed another personality model based on two dimensions: affiliation and dominance (Figure 4). They show that different personality types can be considered points around a circular structure formed in two-dimensional space. The smaller number of dimensions allows them to be controlled more effectively and independently. Two parameters are also easier to visualize, perceive, and understand.

The perception of personality type and traits based on observation has long been a subject of research in behavioral psychology [8–10, 25]. Unfortunately, this research has not focused on facial actions, and has primarily considered the observation of full-body behaviors. Also, mainly due to logistical reasons, the observations have been mostly limited to photographs or few dynamic actions. High-quality and controllable animated characters have not been available to psychology researchers. As Borkenau et al. [9, 10] have illustrated, viewers can achieve relatively stable perceptions using short videos. Creating videos of live actors playing many different and configurable actions, however, can be expensive and difficult.

Among facial actions, the universal facial expressions of emotions (joy, sadness, anger, fear, surprise, and disgust, as described by Ekman [18]) is the only group whose effect on the perception of personality has been investigated. Knutson [25] reported on the effect of facial expression of emotions on interpersonal trait inference based on Wiggins' model. He concludes that viewers attribute high dominance and affiliation to individuals with happy expressions, high dominance, and low affiliation to those with angry or disgusted expressions, and low dominance to those with fearful or sad expressions. Borkenau and Liebler [10] have reported one of the few studies which explicitly associated body gestures and behaviors as visual cues to the perception of personality. They have also considered audio and visual (static and dynamic) cues but facial actions were not a major focus.

2.3. Believable social agents

Badler et al. [4] proposed one of the first personality models or agents to control behavior (in their case, locomotion) based on certain individual characteristics. The proposed architecture includes a physical movement layer, a state machine for behavioral control, and an agent layer that configures the parameters of the state machine. The model is not linked to any theoretically sound personality model, and is a general architecture for configurable behavioral controllers. Other researchers (e.g., [29]) have also proposed methods for modeling agent behaviors. Among them, Rousseau and

Hayes-Roth [36] define behavior as a combination of personality, mood, and attitude. The idea of separating independent components of behavior can be very helpful in designing autonomous agents. Funge et al. [20], on the other hand, propose the idea of hierarchical modeling, which includes behavioral and cognitive modeling layers at the top.

Another approach in behavioral modeling for agents includes associating different facial actions with certain states and events. Cassell et al.[13] propose a method for automatically suggesting and generating facial expressions and some other gestures based on the contents of the speech. In a later work, Cassell et al. [14] propose a comprehensive toolkit with a dedicated language for generating movements based on speech, through certain configurable rules. King et al. [24] and Smid et al. [38] (among others) provide more recent examples of the automatic generation of facial actions (primarily expressions) based on speech. The main weakness of all these works is that the facial actions are (1) usually limited to the expressions, and (2) speech, and not a personality model, is the base for facial actions. A system to suggest facial actions based on personality settings has not been fully investigated.

Associating facial actions with personality requires a reasonably adequate personality model for the agent, and a thorough study of the effect of facial actions on the perception of personality. The latter, as mentioned before, has not been done properly yet, but the former has been the subject of some recent works. Kshirsagar and Magnenat-Thalmann [26] propose a multilayer personality model. It is, more precisely, a multilayer behavioral model that includes layers of personality, mood, and emotions on top of each other. Every layer controls the one below it, with the facial actions and expressions at the bottom. The model allows definition of parameters at each level to individualize the agent. At the personality level, it utilizes the Big-5 model with five parameters. The following observations can be made regarding this system.

(i) The general issues with Big-5.
(ii) Hierarchical dependence of emotional states to personality. The likelihood of transition between emotions can be a personality parameter, but emotional state should be also independently controllable regardless of personality.
(iii) Lack of direct link between facial actions and personality. Speech content or a probabilistic belief networks are used to control facial actions, which may not be enough. Ideally, the facial actions (e.g., the way an agent moves his/her head or raises eye brows and how frequently he/she does it) need to be controlled by a well-defined personality type, entirely or together with speech and likelihood settings (see Section 5 for more details).
(iv) Unnecessary separation of moods and emotions (see Section 6 for a more detailed discussion of moods and emotions).

Models proposed by Egges et al. [17] and Pelachaud and Bilvi [33] follow similar ideas. The latter uses a two-dimensional model similar to Wiggins et al. [40] for personality (called performatives) and also separates them from emotions as two independent components activating facial actions through a belief network. The high-level personality parameters are associated to facial actions based on limited observation and arbitrary settings, rather than a well-performed user study. On the other hand, the facial actions are not limited to speech and can occur even when the agent is not talking, but they have to be set explicitly where desired, while the ideal situation is to define them as part of a personality to be activated autonomously.

2.4. Facial expression of emotions

Russell [37] has mapped emotional states onto a two-dimensional space controlled by arousal and valence. The detailed study of facial actions involved in the expression of the six universal emotions [18] has helped the computer graphics community to develop realistic facial animations. Yet the rules by which these facial expressions are combined to convey more subtle information remain less well understood by behavioral psychologists and animators. This lack of a strong theoretical basis for combining facial actions has resulted in the use of ad-hoc methods for blending facial expression in animations [27, 31, 32, 34]. These methods are usually based on a "weighted average" of facial actions caused by each expression. They are therefore computationally tractable, but the question of their "perceptual" and "psychological" validity has not yet been answered.

3. MULTISPACE BEHAVIORAL MODEL

In the previous section, we reviewed some of the related works in the area of behavioral modeling. Considering the strengths and weaknesses of these approaches, the authors have concluded that the following features are required for a comprehensive agent behavior model. It appears that none of the existing approaches provides a complete collection of them.

(i) A behavioral model needs to be based on scientific findings and models in behavioral psychology.
(ii) The model should have easy-to-visualize parameters for character design.
(iii) The model should consist of separate modules for different behavioral aspects such as knowledge, personality traits, and emotions.
(iv) These behavioral modules should be independent but able to interact with each other and with the underlying geometry.
(v) The parameter spaces and the scripting language should be MPEG-4 compatible.
(vi) The language has to support dynamic actions and interactive scenarios through proper decision making and event handling.

Based on these guidelines, and especially using the suggested model by Rousseau and Hayes-Roth [36], we propose a multispace behavioral model formed with four independent but interacting parameter spaces: geometry, knowledge,

personality, and mood. We replace Rousseau and Hayes-Roth's attitude component with knowledge which includes tasks to be performed and rules of behavior and can provide a better control over agent actions. We also define these four components as parameter spaces formed with specific easily adjustable parameters. These parameter spaces are used in our comprehensive facial animation system, iFACE [2].

iFACE geometry is a hierarchical model that isolates details such as vertex/pixel information from higher-level constructs such as feature and head component, so that animation can be designed and controlled independent of the underlying geometry type. The main advantages of our knowledge space are specially designed language for facial animation, support for decision making and dynamic actions, and high-level timing control. The personality and mood spaces use current findings in behavioral psychology to relate personality traits and emotional states to facial actions, to cause the perception of intended personality type or create the perceptually valid expression. Unlike Kshirsagar and Magnenat-Thalmann's model [26], they perform in total independence from each other (i.e., parameters set separately), but the personality parameters can also define some mood-related aspects of behavior such as the likelihood of transition between emotional states which is in fact a personality based issue (although mood settings can override personality settings temporarily). The mood space does not have any direct effect on personality settings which is again based on "real world" relationships between personality and mood. These spaces are explained in the following sections.

4. HIERARCHICAL GEOMETRY SPACE

Head/face components and regions allow grouping of head data into parts that perform specific actions together (e.g., resizing the ears or closing the eye), which results in isolating details from higher-level commands. This is a key concept in designing an efficient head model. By defining different layers of abstraction on top of actual head data (2D pixels or 3D vertices), each exposing proper interfaces for possible commands, we allow programmers/animators to access only the desired level of details, as illustrated in Figure 5. At the same time, this hierarchy allows changes in lower-level modules (e.g., the way movement of lip corner affects neighbouring points) without any change in the general behavior of higher-level parameters (e.g., an expression can still result in lip corner stretching without a need to know how that happens). Possibility of working with different types of 2D and 3D head data, using the same parameters, is another advantage of such isolation.

Features are special lines/areas that lead facial actions, and feature points (corresponding to MPEG-4 parameters) are control points located on features. Only the lowest level (physical point) depends on the actual (2D or 3D) data. iFACE geometry object model corresponds to this hierarchy and exposes proper interfaces and parameters for client programs to access only the required details for each action. iFACE authoring tool (iFaceStudio) allows users to select feature points and regions-of-influence for them. Each level of geometry accesses the lower levels internally, hiding the details from users and programmers. Eventually, all the facial actions are performed by applying MPEG-4 FAPs to the face.

5. PARAMETERIZED PERSONALITY SPACE

The primary objective of personality modeling is to make it possible for the agent to perform facial actions that cause the viewer to perceive certain personality types, as intended by the character designer. As discussed in Section 2, Wiggins' circumplex model provides an effective parameterized framework for modeling and defining personality types. On the other hand, the effect of dynamic facial actions on personality perception has not been studied properly, partly due to difficulty of hiring actors to record variety of head and face movements [3, 8–10]. Using a realistic facial animation system can help researchers to perform a wider range of experiments.

In order to design a perceptually valid personality model (i.e., one that initiates actions that most likely cause the intended personality perception in viewers), we performed a four-step process.

(1) Define sets of facial actions and expressions that may affect personality perception (visual cues).
(2) Run experiments with a large enough user base to study the effect of these visual cues on personality perception.
(3) Associate visual cues to personality parameters, affiliation and dominance.
(4) Create a model that defines parameterized personality profiles and initiates proper facial actions based on that.

Table 1 shows the visual cues selected at step 1 and the results of our experiments with 31 undergraduate students at the Department of Psychology, University of British Columbia, Vancouver, Canada. Details of experiments have been published in an earlier paper [3].

The personality model controls strength and the timing of initiating facial actions based on personality settings. We give each personality parameter three linguistic values: low, medium, and high. For example, for parameter dominance these correspond to dominant, neutral, and submissive, as shown in Figure 4. After performing the experiments, visual cues are associated with each one of these parameter values, to form sets like the following: $C_{i,j} = \{c_{i,j,n}\}$, where $c_{i,j,n}$ is the nth visual cue associated with the jth value of ith parameter.

Each visual cue is defined as an individual MPEG-4 FAP [6] or a combination of them. If p_i is the value of ith personality parameter ($i = 0$ *or* 1), $v_{i,j}$ (the strength of the jth linguistic values of that parameter) will be calculated using a fuzzy membership function based on p_i. These strengths are then used to activate the visual cues to certain levels:

$$a_{i,j,n} = v_{i,j} \times m_{i,j,n} \tag{1}$$

$a_{i,j,n}$ and $m_{i,j,n}$ are activation level of the visual cue (or the related FAP) and its maximum value, respectively.

Table 1: Affiliation and dominance scores for facial actions (min = −5, max = 5).

Facial action	Affiliation	Dominance
Joy	4.7	2
Sadness	0.2	−0.2
Anger	−2.6	0.6
Fear	2	−0.8
Disgust	0.9	1
Surprise	2.9	−1
Contempt	−5.7	1.4
Neutral	0.8	−0.8
Slow turn	1.7	1.2
Slow tilt	0.9	0.2
Slow nod	−0.5	−3.1
Slow blink	2.5	−0.7
Slow avert	0.1	−0.7
Slow one brow	−0.1	0
Slow two brows	4.2	−0.9
Fast turn	2.5	1.7
Fast tilt	2.1	1.9
Fast nod	−0.6	−2.8
Fast blink	2.7	−0.8
Fast avert	−0.2	−2.9
Fast one brow	−1.6	3.3
Fast two brows	3.8	0.9
Head rest down	−0.1	−1.4
Head rest side	0.4	−3.4

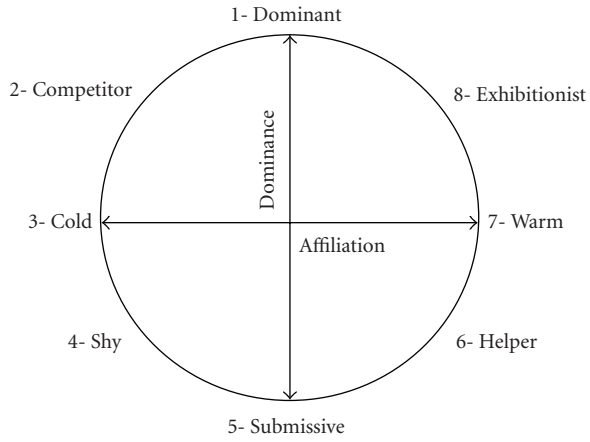


Figure 4: Wiggins' personality circumplex.

The timing for activating visual cues is also set in the personality profile. It can be random, periodic, or based on speech energy level. The content of the speech can also be used as suggested by other researchers [38]. Some measures of speech energy can be calculated by analyzing the speech signal. Two strength thresholds of impulse and emphasis can be defined for this energy. Different visual cues (or different versions of them with varied maximum values) can be associated with these thresholds. Once a threshold is reached,

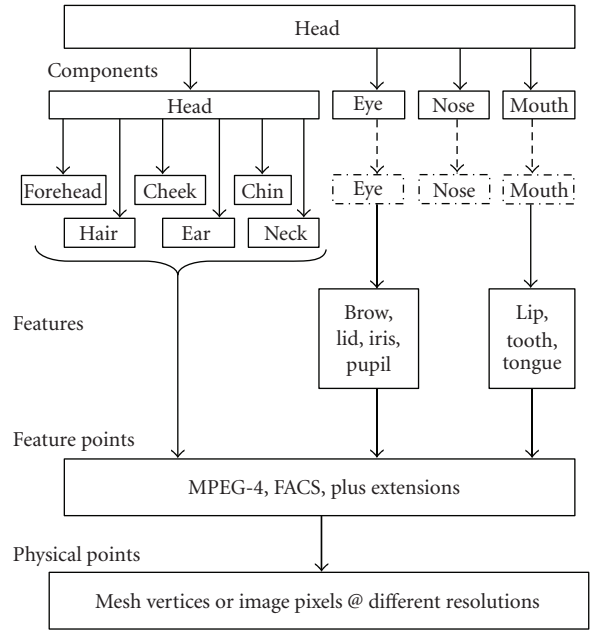


Figure 5: Hierarchical facial geometry.

one of the associated cues that matches the agent personality is randomly selected and activated based on the value of $a_{i,j,n}$.

6. PARAMETERIZED MOOD SPACE

The distinction between moods and emotions has been discussed by many researchers. The major differences seem to be duration and cause, and the emotions are believed to be more external and visible [7]. Due to complicated relation between moods and emotions, and between moods and visual appearance, it is hard to create mood parameters (independent of emotions) that can effectively and clearly control the facial actions. Some researchers [26] have tried to define such parameters for an agent's mood in which the result is simply three types of moods (bad, normal, and good) which only change the likelihood of transition between emotions and have no extra functionality (e.g., direct effect on facial actions).

In our model, we consider emotions and mood part of one parameter space called mood. This space controls the emotional state through two parameters (see Figure 6), and also includes probability settings for random or event-based transition between emotions. With better understanding of how moods affect emotions and other visual aspects, we hope to separate moods and emotions into two parameter spaces, but at this time a simple "likelihood setting" does not seem enough for such separation.

The emotional state of the agent can be set in three different ways.

(1) Explicitly in the course of an action (see FML scripts).
(2) Randomly/periodically as configured in the personality profile.

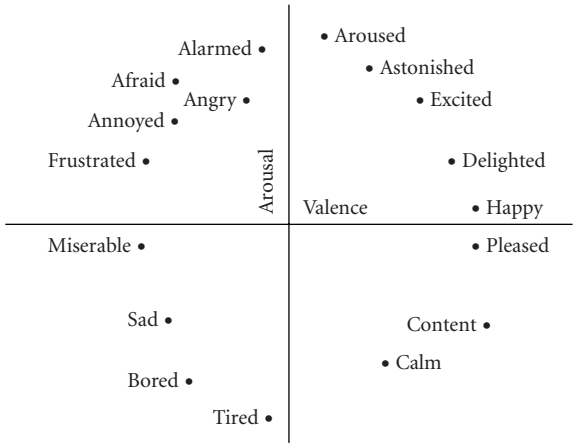


Figure 6: Parameterized mood space [37].

(3) Randomly/periodically as configured in the mood space which overrides personality setting.

In either case, the mood (or emotional state) is set by specifying a universal emotion and its level of activation, or by setting the values of two mood parameters: valence and arousal (see Figure 6). Ekman has described the facial actions associated with the expression of universal emotions in detail [18]. For example, the expression of joy involves tightening of eyelids, raising cheeks, lowering eyebrows slightly, and wrinkles around the eyes especially the corners. For single universal emotions, we activate the associated actions based on the level of emotional state. For blending two expressions, we differentiate between two cases: transition from one expression to another, and activation of two expressions at the same time, that is, a combined expression.

The facial actions for transitions are simply the weighted average of the source and destination expressions:

$$a_i = k \times a_{i,s} + (1 - k) \times a_{i,d}, \qquad k = \frac{N - f}{N}. \tag{2}$$

N is the number of frames to create for the transition, f is the current frame, a_i is the activation of ith action at frame f, and $a_{i,s}$ and $a_{i,d}$ are the activation of that action in source and destination expressions.

The combined expressions are created by either selecting two universal expressions, or by setting arousal and valence parameters. In the first case, the activation levels of two expressions are first mapped into a pair of arousal-valence parameters. The resulting values of arousal and valence are then used to activate facial actions associated with each parameter as shown in Table 2. These facial actions are selected by analyzing the Ekman's description of universal expressions and their facial actions, and by clustering similar actions based on arousal and valence parameters.

These two cases are illustrated in Figure 7. In this figure, (a) and (b) show surprise and anger expressions. The middle frame for transition (c) is between (a) and (b). We see that due to the raised mid lower lip in anger (target of linear interpolation), the middle of the mouth closes while the sides are not closed yet. This may be acceptable for a transition but in case of a combined expression like aroused, it is better to locate the source and target on arousal-valence map, and then find the proper (perceptually valid) facial actions for a point between them. This is shown in (d) where the jaw is slightly dropped, upper and lower eyelids are raised a little, and brows are slightly lowered and drawn together. The effectiveness of the parameter-based expression blending compared to the simple weighted average method is the subject of an extensive user study in the University of British Columbia. The details of this study will be presented in a separate paper.

7. FACE MODELING LANGUAGE

7.1. Design ideas

To describe the tasks to be performed, the timing, and event handling mechanism, a special-purpose language for facial animation has been designed for iFACE that performs proper configuration and controls the main sequence of actions. The need for such a high-level language, as opposed to low-level parameters such as those in MPEG-4, can be shown using an example. Figure 8 illustrates a series of facial actions. A "wink" (closing eye lid and lowering eyebrow), a "head rotation," and a "smile" (only stretching lip corners, for simplicity). These actions can be described by the following MPEG-4 FAPs.

Wink: FAP-31 (raise-l-i-eyebrow),
FAP-33 (raise-l-m-eyebrow),
FAP-35 (raise-l-o-eyebrow),
FAP-19 (close-t-l-eyelid).

Head rotation:
FAP-49 (head rotation -yaw).

Smile: FAP-6 (stretch-l-lipcorner),
FAP-6 (stretch-r-lipcorner).

Although simple and powerful, the use of MPEG-4 FAPs for behavioral description lacks the following features.

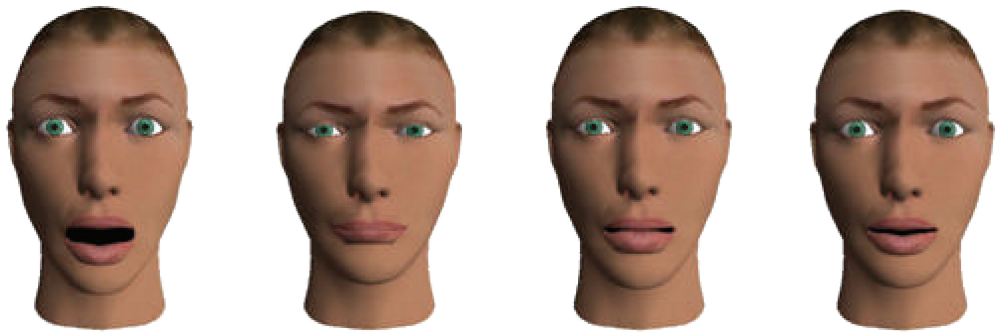
(1) Parameters at facial component level (e.g., one eye wink instead of four FAPs).
(2) Proper timing mechanism (e.g., duration and dependencies).
(3) Event handling and decision-making.

Face modeling language (FML) [2] is an XML-based language designed for facial animation. It combines MPEG-4 compatibility with higher-level features such as those mentioned above. Also, FML is independent of the underlying animation system. The actions of Figure 8 can be done by an FML script such as lines shown in Figure 9 (elements are discussed later).

FML defines a timeline of events (Figure 10) including head movements, speech, and facial expressions, and their combinations. Temporal combination of facial actions is

Table 2: Sample facial actions and the expressions that include them.

Action	Expressions	Valence	Arousal
Brows drawn together	Fear, anger	Low	High
Brows lowered	Joy, anger, disgust	—	High
Brows raised	Fear, surprise	—	High
Brows-inner raised	Sadness	Low	Medium
Eye-corner wrinkled	Joy	High	—
Eye-lid-lower raised	Joy, sadness	—	Medium/low
Eye-lid-lower tensed and raised	Fear, anger	Low	High
Eye-lid-upper lowered	Joy, sadness	—	Medium/low
Eye-lid-upper raised	Fear, anger, surprise	—	High
Jaw dropped	Surprise	Medium	High
Jaw thrusted forward	Anger	Low	High
Lip-corners lowered	Sadness	Low	Low
Lip-corners raised	Joy	High	Medium
Lip-lower raised	Sadness	Low	Low
Lips pressed and narrowed	Anger	Low	High
Lips stretched	Joy, fear, anger	—	Medium



(a) (b) (c) (d)

Figure 7: Samples of expression blending: (a) surprise, (b) anger, (c) transition between surprise and anger, and (d) blending based on valence and arousal and their associated facial actions.

done through time containers which are XML tags borrowed from SMIL (other language elements are FML specific). Since a face animation might be used in an interactive environment, such a timeline may be altered/determined by a user. So another functionality of FML is to allow user interaction and in general event handling (decision making based on external events and dynamic generation of scenarios).

7.2. FML document structure

An FML document consists, at the higher level, of two types of elements: `model` and `story`. A `model` element is used for defining face capabilities, parameters, and initial configuration. This element groups other FML elements (model items) such as configuration data and predefined actions. A `story` element, on the other hand, represents the timeline of events in face animation in terms of individual actions (FML `action` elements). The face animation timeline consists of facial activities and their temporal relations. These activities are themselves sets of simple "moves." Sets of these moves are grouped together within "time containers," that is, special XML tags that define the temporal relationships of the elements inside them. FML includes three SMIL time containers: `excl`, `seq`, and `par` representing exclusive, sequential and parallel move sets. Other XML tags are specifically designed for FML.

FML supports three basic face moves: talking, expressions, and 3D head movements. Combined through time containers, they form an FML `action` which is a logically related set of activities. Details of these moves and other FML elements and constructs will be discussed in the next subsections. The special `fap` and `param` elements are also included for MPEG-4 FAPs and other system-dependent parameters. Time containers are FML elements that represent the temporal relation between moves. The basic time containers are `seq` and `par` corresponding to sequential and

(a) (b) (c) (d)

Figure 8: Series of facial actions: (a) start, (b) wink, (c) head rotation, (d) smile.

```
<seq>
<param type=‘‘comp’’ name=‘‘eye-wink’’
       duration=‘‘1s’’/>
<hdmv type=‘‘yaw’’ value=‘‘20’’
       duration=‘‘1s’’/>
<expr type=‘‘smile’’ value =‘‘50’’
       duration=‘‘1s’’/>
</seq>
```

Figure 9: FML script for actions in Figure 8.

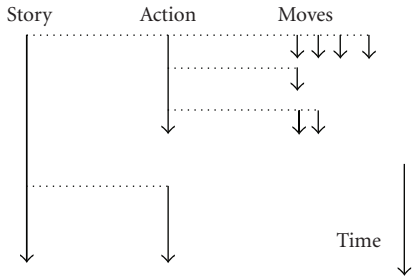


Figure 10: FML timeline and temporal relation of face activities.

```
<action>
    <seq begin=‘‘0’’>
        <talk>Hello</talk>
        <hdmv end=‘‘5s’’ type=‘‘0’’
           val=‘‘30’’ />
    </seq>
    <par begin =‘‘0’’>
       <talk>Hello</talk>
       <expr end=‘‘3s’’ type =‘‘3’’
           val=‘‘50’’ />
    </par>
</action>
```

Figure 11: Time containers and basic moves.

```
<!-- in model part -->
<event name=‘‘user’’ val=‘‘-1’’ />

<!-- in story part -->
<excl ev_name=‘‘user’’>
       <talk ev_val=‘‘0’’>Hello</talk>
       <talk ev_val=‘‘1’’>Bye</talk>
</excl>
```

Figure 12: Decision making and event handling.

parallel activities. The former contains moves that start one after another, and the latter contains moves that begin at the same time. Time containers include primitive moves and also other time containers in a nested way. The `repeat` attribute of the time container elements allows iteration in FML documents as illustrated later in sample applications.

Similar to SMIL, FML also has a third type of time containers, `excl`, used for implementing exclusive activities and decision making as discussed later. All `story` elements have four timing attributes: `repeat`, `begin`, `duration`, and `end`. In a sequential time container, `begin` is relative to start time of the previous move, and in a parallel container it is relative to the start time of the container. In case of a conflict, duration of moves is set according to their own settings rather than the container. The `repeat` attribute is considered for defining definite (when having an explicit value) or indefinite loops (associated with events). FML time containers and basic moves are illustrated in Figure 11.

7.3. Event handling and decision making

In dynamic and interactive applications, the FML document needs to make decisions, that is, to follow different paths based on certain events. To accomplish this, `excl` time container and `event` element are added. An event represents any external data, for example, the value of a user selection. The new time container associates with an `event` and allows waiting until the `event` has one of the given values, then it continues with exclusive execution of the action corresponding to that value, as illustrated in Figure 12. The system component processing FML scripts exposes proper interface function to allow event values to be set in run time. `event` is the FML counterpart of familiar if-else constructs in normal programming languages.

```
<!-- in model part -->
<event name=‘‘userChoice’’ val=‘‘-1’’ />
<param name=‘‘dominance’’ val=‘‘60’’ />
<param name=‘‘affiliation’’ val=‘‘90’’ />
<data name=‘‘reply-1’’ val=‘‘Hello’’ />
<data name=‘‘reply-2’’ val=‘‘Fine’’ />

<!-- in story part -->
<excl ev_name=‘‘userChoice’’
        repeat=‘‘userChoice;4’’>
    <talk ev_val=‘‘1’’ name=‘‘reply-1’’>
         </talk>
    <talk ev_val=‘‘2’’ name=‘‘reply-2’’>
         </talk>
    <talk ev_val=‘‘3’’ name=‘‘reply-3’’>
         </talk>
</excl>
```

Figure 13: FML script for interactive agent.

8. SAMPLE APPLICATIONS

In this section, we review sample application using iFACE system and our proposed behavioral model. For more information, sample applications, and videos please see our research web site http://ivizlab.sfu.ca/research.

8.1. Interactive agent

Typical examples of an interactive agent are game characters and online customer service representatives. In such cases, the agent needs to follow a main scenario, allow nonlinear sequences of events (e.g., making a decision based on a user input and going through different paths as the result), show emotions, and have a certain personality. Figure 13 demonstrates a sample FML script for such an agent.

This script creates a character that waits for user questions and replies to them. The user interface is controlled by the GUI application. It provides four options: “Hello,” “How are you?” a user-typed question, and “Bye.” The reply to options 1 and 2 are hard coded in the script (data elements in model). The reply to the third (user-typed) question will be provided by the background application (i.e., the intelligence behind the script). The fourth user option ends the script.

In the model part of the script, the personality parameters are set, a user event has been declared and set to −1 (default value, meaning not defined), and finally two data items have been set for user options 1 and 2. The main actions are controlled in the `excl` element. The repeat attribute defines the ending condition. The `excl` options look for the appropriate reply, either in the script or from the background application (through the iFACE API not shown here).

Table 3: Example relations between music features and emotions [11, 23, 28].

Emotion	Feature	Value
Fear	Tempo	Irregular
	Sound level	Low
	Articulation	Mostly nonlegato
Anger	Tempo	Very rapid
	Sound level	Load
	Articulation	Mostly nonlegato
Happiness	Tempo	Fast
	Sound level	Moderate or load
	Articulation	Airy

Figure 14: Sample animated heads from MusicFace.

8.2. MusicFace

Music-driven emotionally expressive face (MusicFace) [16] is a multimedia application based on iFACE to demonstrate the concept of affective communication remapping, that is, transforming affective information from one communication medium to another. Affective information is extracted from a piece of music by analyzing musical features such as rhythm, energy, timbre, articulation, and melody (see Table 3).

After setting general personality type and parameters based on the music, the emotional state is determined and updated continuously using the following algorithm (sample animation frames in Figure 14).

(1) Select high or low arousal emotions based on music power level.
(2) Select positive or negative valence emotions based on timbre and rhythm.
(3) Fine tune emotional state based on other musical features.

9. CONCLUSION

We have described a behavioral model for social agents that consists of four independent but interacting parameter spaces: geometry, knowledge, personality, and mood. Personality and mood are modeled based on current findings in behavioral psychology, relating the perception of personality and the emotional states to facial actions and expressions. The character knowledge and tasks to be performed, in addition to the rules of behavior and decision making, are encapsulated in a specially designed language that is also

compatible with the MPEG-4 standard. Associating facial actions to parameters (affective or personality dimensions) rather than "basic emotions" or "personality types" allows a designer to easily change the parameters and create new personality types and combined expressions that are perceptually valid. Further research is needed to study the effect of cultural background on such perception.

REFERENCES

[1] Y. Arafa, K. Kamyab, E. Mamdani, et al., "Two approaches to scripting character animation," in *Proceedings of the 1st International Conference on Autonomous Agents & Multi-Agent Systems, Workshop on Embodied Conversational Agents*, Bologna, Italy, July 2002.

[2] A. Arya, S. DiPaola, L. Jefferies, and J. T. Enns, "Socially communicative characters for interactive applications," in *Proceedings of the 14th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision (WSCG '06)*, Plzen-Bory, Czech Republic, January-February 2006.

[3] A. Arya, L. N. Jefferies, J. T. Enns, and S. DiPaola, "Facial actions as visual cues for personality," *Computer Animation and Virtual Worlds*, vol. 17, no. 3-4, pp. 371–382, 2006.

[4] N. Badler, B. D. Reich, and B. L. Webber, "Towards personalities for animated agents with reactive and planning behaviors," in *Creating Personalities for Synthetic Actors: Towards Autonomous Personality Agents*, R. Trappl and P. Petta, Eds., pp. 43–57, Springer, New York, NY, USA, 1997.

[5] J. Bates, "The role of emotion in believable agents," *Communications of the ACM*, vol. 37, no. 7, pp. 122–125, 1994.

[6] S. Battista, F. Casalino, and C. Lande, "MPEG-4: a multimedia standard for the third millennium—part 1," *IEEE Multimedia*, vol. 6, no. 4, pp. 74–83, 1999.

[7] C. J. Beedie, P. C. Terry, and A. M. Lane, "Distinctions between emotion and mood," *Cognition and Emotion*, vol. 19, no. 6, pp. 847–878, 2005.

[8] D. S. Berry, "Accuracy in social perception: contributions of facial and vocal information," *Journal of Personality and Social Psychology*, vol. 61, no. 2, pp. 298–307, 1991.

[9] P. Borkenau, N. Mauer, R. Riemann, F. M. Spinath, and A. Angleitner, "Thin slices of behavior as cues of personality and intelligence," *Journal of Personality and Social Psychology*, vol. 86, no. 4, pp. 599–614, 2004.

[10] P. Borkenau and A. Liebler, "Trait inferences: sources of validity at zero acquaintance," *Journal of Personality and Social Psychology*, vol. 62, no. 4, pp. 645–657, 1992.

[11] R. Bresin and A. Friberg, "Synthesis and decoding of emotionally expressive music performance," in *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, vol. 4, pp. 317–322, Tokyo, Japan, October 1999.

[12] D. C. A. Bulterman, "SMIL 2.0—part 1: overview, concepts, and structure," *IEEE Multimedia*, vol. 8, no. 4, pp. 82–88, 2001.

[13] J. Cassell, C. Pelachaud, N. Badler, et al., "Animated conversation: rule-based generation of facial expression, gesture and spoken intonation for multiple conversational agents," in *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '94)*, pp. 413–420, New York, NY, USA, July 1994.

[14] J. Cassell, H. H. Vilhjálmsson, and T. Bickmore, "BEAT: the behaviour expression animation toolkit," in *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '01)*, pp. 477–486, Los Angeles, Calif, USA, August 2001.

[15] B. De Carolis, C. Pelachaud, I. Poggi, and M. Steedman, "APML, a markup language for believable behaviour generation," in *Proceedings of the 1st International Conference on Autonomous Agents & Multi-Agent Systems, Workshop on Embodied Conversational Agents*, Bologna, Italy, July 2002.

[16] S. DiPaola and A. Arya, "Affective communication remapping in musicface system," in *Proceedings of the 10th European Conference on Electronic Imaging and the Visual Arts (EVA '04)*, London, UK, July 2004.

[17] A. Egges, S. Kshirsagar, and N. Magnenat-Thalmann, "A model for personality and emotion simulation," in *Proceedings of the 7th International Conference on Knowledge-Based Intelligent Information & Engineering Systems (KES '03)*, pp. 453–461, Oxford, UK, September 2003.

[18] P. Ekman, *Emotions Revealed*, Consulting Psychologists Press, San Francisco, Calif, USA, 1978.

[19] P. Ekman and W. V. Friesen, *Facial Action Coding System*, Consulting Psychologists Press, San Francisco, Calif, USA, 1978.

[20] J. Funge, X. Tu, and D. Terzopoulos, "Cognitive modeling: knowledge, reasoning and planning for intelligent characters," in *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '99)*, pp. 29–38, Los Angeles, Calif, USA, August 1999.

[21] L. R. Goldberg, "An alternative "description of personality": the big-five factor structure," *Journal of Personality and Social Psychology*, vol. 59, no. 6, pp. 1216–1229, 1990.

[22] C. Jones, *Chuck Amuck : The Life and Times of Animated Cartoonist*, Farrar, Straus, and Giroux, New York, NY, USA, 1989.

[23] P. N. Juslin, "Cue utilization in communication of emotion in music performance: relating performance to perception," *Journal of Experimental Psychology: Human Perception and Performance*, vol. 26, no. 6, pp. 1797–1813, 2000.

[24] S. A. King, A. Knott, and B. McCane, "Language-driven nonverbal communication in a bilingual conversational agent," in *Proceedings of the 16th International Conference on Computer Animation and Social Agents (CASA '03)*, pp. 17–22, New-Brunswick, NJ, USA, May 2003.

[25] B. Knutson, "Facial expressions of emotion influence interpersonal trait inferences," *Journal of Nonverbal Behavior*, vol. 20, no. 3, pp. 165–181, 1996.

[26] S. Kshirsagar and N. Magnenat-Thalmann, "A multilayer personality model," in *Proceedings of the 2nd International Symposium on Smart Graphics*, pp. 107–115, Hawthorne, NY, USA, June 2002.

[27] W.-S. Lee, M. Escher, G. Sannier, and N. Magnenat-Thalmann, "MPEG-4 compatible faces from orthogonal photos," in *Proceedings of Computer Animation (CA '99)*, pp. 186–194, Geneva, Switzerland, May 1999.

[28] D. Liu, L. Lu, and H.-J. Zhang, "Automatic mood detection from acoustic music data," in *Proceedings of the 4th International Symposium on Music Information Retrieval (ISMIR '03)*, Baltimore, Md, USA, October 2003.

[29] A. B. Loyall and J. B. Bates, "Personality-rich believable agents that use language," in *Proceedings of the 1st International Conference on Autonomous Agents*, pp. 106–113, Marina del Rey, Calif, USA, February 1997.

[30] A. Marriott and J. Stallo, "VHML: uncertainties and problems. A discussion," in *Proceedings of the 1st International Conference*

on Autonomous Agents & Multi-Agent Systems, Workshop on Embodied Conversational Agents, Bologna, Italy, July 2002.

[31] J.-Y. Noh and U. Neumann, "Expression cloning," in *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '01)*, pp. 277–288, Los Angeles, Calif, USA, August 2001.

[32] A. Paradiso, "An algebra of facial expressions," in *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '00)*, New Orleans, La, USA, July 2000.

[33] C. Pelachaud and M. Bilvi, "Computational model of believable conversational agents," in *Communication in Multiagent Systems: Background, Current Trends and Future*, M.-P. Huget, Ed., pp. 300–317, Springer, New York, NY, USA, 2003.

[34] K. Perlin, "Layered compositing of facial expression," in *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '97)*, Los Angeles, Calif, USA, August 1997.

[35] H. Prendinger, S. Descamps, and M. Ishizuka, "Scripting affective communication with life-like characters in web-based interaction systems," *Applied Artificial Intelligence*, vol. 16, no. 7-8, pp. 519–553, 2002.

[36] D. Rousseau and B. Hayes-Roth, "Interacting with personality-rich characters," Report KSL 97-06, Knowledge Systems Laboratory, Stanford University, Stanford, Calif, USA, 1997.

[37] J. A. Russell, "A circumplex model of affect," *Journal of Personality and Social Psychology*, vol. 39, no. 6, pp. 1161–1178, 1980.

[38] K. Smid, I. Pandzic, and V. Radman, "Autonomous speaker agent," in *Proceedings of Computer Animation and Social Agents Conference (CASA '04)*, Geneva, Switzerland, July 2004.

[39] D. Watson, "Strangers' ratings of the five robust personality factors: evidence of a surprising convergence with self-report," *Journal of Personality and Social Psychology*, vol. 57, no. 1, pp. 120–128, 1989.

[40] J. S. Wiggins, P. Trapnell, and N. Phillips, "Psychometric and geometric characteristics of the revised interpersonal adjective scales (IAS-R)," *Multivariate Behavioral Research*, vol. 23, no. 3, pp. 517–530, 1988.

Hindawi Publishing Corporation
EURASIP Journal on Image and Video Processing
Volume 2007, Article ID 29081, 22 pages
doi:10.1155/2007/29081

Research Article

Robust Feature Detection for Facial Expression Recognition

Spiros Ioannou, George Caridakis, Kostas Karpouzis, and Stefanos Kollias

Image, Video and Multimedia Systems Laboratory, National Technical University of Athens, 9 Iroon Polytechniou Street, 157 80 Zographou, Athens, Greece

Received 1 May 2006; Revised 27 September 2006; Accepted 18 May 2007

Recommended by Jörn Ostermann

This paper presents a robust and adaptable facial feature extraction system used for facial expression recognition in human-computer interaction (HCI) environments. Such environments are usually uncontrolled in terms of lighting and color quality, as well as human expressivity and movement; as a result, using a single feature extraction technique may fail in some parts of a video sequence, while performing well in others. The proposed system is based on a multicue feature extraction and fusion technique, which provides MPEG-4-compatible features assorted with a confidence measure. This confidence measure is used to pinpoint cases where detection of individual features may be wrong and reduce their contribution to the training phase or their importance in deducing the observed facial expression, while the fusion process ensures that the final result regarding the features will be based on the extraction technique that performed better given the particular lighting or color conditions. Real data and results are presented, involving both extreme and intermediate expression/emotional states, obtained within the sensitive artificial listener HCI environment that was generated in the framework of related European projects.

1. INTRODUCTION

Facial expression analysis and emotion recognition, a research topic traditionally reserved for psychologists, has gained much attention by the engineering community in the last twenty years. Recently, there has been a growing interest in improving all aspects of the interaction between humans and computers, providing a realization of the term "affective computing." The reasons include the need for quantitative facial expression description [1] as well as automation of the analysis process [2] which is strongly related to ones' emotional and cognitive state [3].

Automatic estimation of facial model parameters is a difficult problem and although a lot of work has been done on selection and tracking of features [4], relatively little work has been reported [5] on the necessary initialization step of tracking algorithms, which is required in the context of facial feature extraction and expression recognition. Most facial expression recognition systems use the facial action coding system (FACS) model introduced by Ekman and Friesen [3] for describing facial expressions. FACS describes expressions using 44 action units (AU) which relate to the contractions of specific facial muscles. In addition to FACS, MPEG-4 metrics [6] are commonly used to model facial expressions and underlying emotions. They define an alternative way of modeling facial expressions and the underlying emotions, which is strongly influenced by neurophysiologic and psychological studies. MPEG-4, mainly focusing on facial expression synthesis and animation, defines the facial animation parameters (FAPs) that are strongly related to the action units (AUs), the core of the FACS. A comparison and mapping between FAPs and AUs can be found in [7].

Most facial expression recognition systems attempt to map facial expressions directly into archetypal emotion categories while been unable to handle expressions caused by intermediate or nonemotional expressions. Recently, several automatic facial expression analysis systems that can also distinguish facial expression intensities have been proposed [8–11], but only a few are able to employ model-based analysis using the FAP or FACS framework [5, 12]. Most existing approaches in facial feature extraction are either designed to cope with limited diversity of video characteristics or require manual initialization or intervention. Specifically, [5] depends on optical flow, [13–17] depend on high resolution or noise-free input video, [18–20] depend on color information, [15, 21] require manual labeling or initialization, [12] requires markers, [14, 22] require manual selection of feature points on the first frame, [23] requires two head-mounted

cameras, [24–27] require per-user or per-expression training either on the expression recognition or the feature extraction or cope only with fundamental emotions. From the above, [8, 13, 21, 23, 25, 27] provide success results solely on expression recognition and not on the feature extraction/recognition. Additionally very few approaches can perform in near real time.

Fast methodologies for face and feature localization in image sequences are usually based on calculation of the skin color probability. This is usually accomplished by calculating the a posteriori probability of a pixel belonging to the skin class in the joint Cb/Cr domain. Several other color spaces have also been proposed which exploit specific color characteristics of various facial features [28]. Video systems, on the other hand, convey image data in the form of one component that represents lightness (luma) and two components that represent color (chroma), disregarding lightness. Such schemes exploit the poor color acuity of human vision: as long as luma is conveyed with full detail, detail in the chroma components can be reduced by subsampling (filtering or averaging). Unfortunately, nearly all video media have reduced vertical and horizontal color resolutions. A 4 : 2 : 0 video signal (e.g., H-261, MPEG-2 where each of Cr and Cb are subsampled by a factor of 2 both horizontally and vertically) is still considered to be a very good quality signal. The perceived video quality is good indeed, but if the luminance resolution is low enough—or the face occupies only a small percentage of the whole frame—it is not rare that entire facial features share the same chrominance information, thus rendering color information very crude for facial feature analysis. In addition to this, overexposure in the facial area is common due to the high reflectivity of the face and color alteration is almost inevitable when transcoding between different video formats, rendering Cb/Cr inconsistent and not constant. Its exploitation is therefore problematic in many real-life video sequences; techniques like the one in [29] have been proposed in this direction but no significant improvement has been observed.

In the framework of the European Information Technology projects, ERMIS [30] and HUMAINE [31], a large audiovisual database was constructed which consists of people driven to emotional discourse by experts. The subjects participating in this experiment were not faking their expressions and the largest part of the material is governed by subtle emotions which are very difficult to detect even for human experts, especially if one disregards the audio signal.

The aim of our work is to implement a system capable of analyzing nonextreme facial expressions. The approach has been tested in a real human-computer interaction framework, using the SALAS (sensitive artificial listener) testbed [30, 31], which is briefly described in the paper. The system should be able to evaluate expressions even when the latter are not extreme and should be able to handle input from various speakers. To overcome the variability in terms of luminance and color resolution in our material, an analytic approach that allows quantitative and rule-based expression profiling and classification was developed. Facial expression is estimated through analysis of MPEG FAPs [32], the latter being measured through detection of movement and deformation of local intransient facial features such as mouth, eyes, and eyebrows through time, assuming availability of a person's neutral expression. The proposed approach is capable of detecting both basic and intermediate expressions (e.g., boredom, anger) [7] with corresponding intensity and confidence levels.

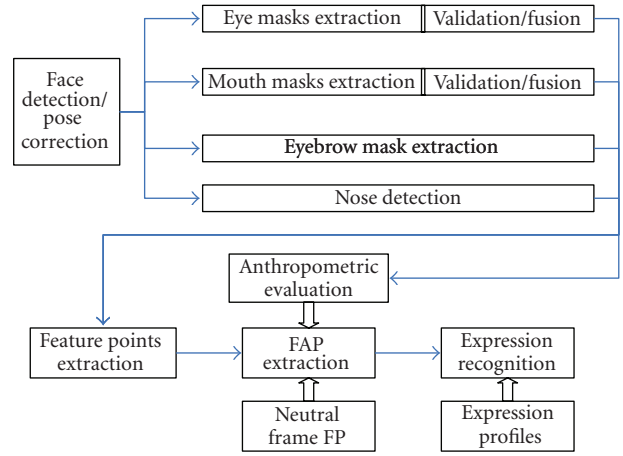


Figure 1: Diagram of the proposed methodology.

An overview of the proposed expression and feature extraction methodologies is given in Section 2 of the paper. Section 3 describes face detection and pose estimation while Section 4 provides detailed analysis of automatic facial feature boundary extraction and construction of multiple masks for handling different input signal variations. Section 5 describes the multiple mask fusion process and confidence generation. Section 6 focuses on facial expression/emotional analysis, and presents the SALAS human-computer interaction framework while Section 7 presents the obtained experimental results. Section 8 draws conclusions and discusses future work.

2. AN OVERVIEW OF THE PROPOSED APPROACH

An overview of the proposed methodology is illustrated in Figure 1. The face is first located, so that approximate facial feature locations can be estimated from the head position and rotation. Face roll rotation is estimated and corrected and the head is segmented focusing on the following facial areas: left eye/eyebrow, right eye/eyebrow, nose, and mouth. Each of those areas, called feature-candidate areas, contains the features whose boundaries need to be extracted for our purposes. Inside the corresponding feature-candidate areas precise feature extraction is performed for each facial feature, that is, eyes, eyebrows, mouth, and nose, using a multicue approach, generating a small number of intermediate feature masks. Feature masks generated for each facial feature are fused together to produce the final mask for that feature. The mask fusion process uses anthropometric criteria [33] to perform validation and weight assignment on each intermediate mask; each feature's weighted masks are then fused to produce a final mask along with confidence level estimation.

Measurement of facial animation parameters (FAPs) requires the availability of a frame where the subject's expression is found to be neutral. This frame will be called the *neutral frame* and is manually selected from video sequences to be analyzed or interactively provided to the system when initially brought into a specific user's ownership. The final feature masks are used to extract 19 feature points (FPs) [7]. Feature points obtained from each frame are compared to FPs obtained from the neutral frame to estimate facial deformations and produce the facial animation parameters (FAPs). Confidence levels on FAP estimation are derived from the equivalent feature point confidence levels. The FAPs are used along with their confidence levels to provide the facial expression estimation.

3. FACE DETECTION AND POSE ESTIMATION

In the proposed approach, facial features including eyebrows, eyes, mouth, and nose are first detected and localized. Thus, a first processing step of face detection and pose estimation is carried out, as described below, to be followed by the actual facial feature extraction process described in Section 4. At this stage, it is assumed that an image of the user at neutral expression is available, either a priori or captured before interaction with the proposed system starts.

The goal of face detection is to determine whether or not there are faces in the image, and if yes, return the image location and extent of each face [34]. Face detection can be performed with a variety of methods [35–37]. In this paper, we used nonparametric discriminant analysis with a *support vector machine* (SVM) which classifies face and nonface areas reducing the training problem dimension to a fraction of the original with negligible loss of classification performance [30, 38].

800 face examples from the NIST Special Database 18 were used for this purpose. All examples were aligned with respect to the coordinates of the eyes and mouth and rescaled to the required size. This set was virtually extended by applying small scale, translation, and rotation perturbations and the final training set consisted of 16 695 examples.

The face detection step provides a rectangle head boundary which includes all facial features as shown in Figure 2. The latter can be then segmented roughly using static anthropometric rules (Figure 2, Table 1) into three overlapping rectangle regions of interest which include both facial features and facial background; these three *feature-candidate areas* include the left eye/eyebrow, the right eye/eyebrow, and the mouth. In the following, we utilize these areas to initialize the feature extraction process. Scaling does not affect feature-candidate area detection, since the latter is proportional to the head boundary extent, extracted by the face detector.

The accuracy of feature extraction depends on head pose. In this paper, we are mainly concerned with roll rotation, since it is the most frequent rotation encountered in real-life video sequences. Small head yaw and pitch rotations which do not lead to feature occlusion do not have a significant impact on facial expression recognition. The face detection techniques described in the former section is able to cope with head roll rotations up to 30°. This is a quite satisfactory

(a) (b)

Figure 2: Feature-candidate areas: (a) full frame (352 × 288), (b) Zoomed (90 × 125).

Table 1: Anthropometric rules for feature-candidate facial areas. W_f, H_f represent face width and face height, respectively.

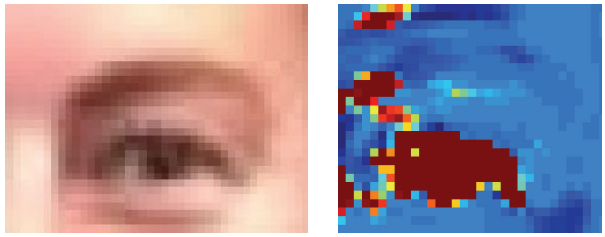
Area	Location	Width	Height
Eyes and eyebrows	Top left and right parts of the face	$0.6W_f$	$0.5H_f$
Nose and mouth	Bottom part of the face	W_f	$0.5H_f$

range in which the feature-candidate areas are large enough so that the eyes reside in the eye-candidate search areas defined by the initial segmentation of a rotated face.

To estimate the head pose, we first locate the left and right eyes in the detected corresponding eye candidate areas. After locating the eyes, we can estimate head roll rotation by calculating the angle between the horizontal plane and the line defined by the eye centers. For eye localization, we propose an efficient technique using a feed-forward backpropagation neural network with a sigmoidal activation function. The multilayer perceptron (MLP) we adopted employs Marquardt-Levenberg learning [39, 40] while the optimal architecture obtained through pruning has two 20-node hidden layers and 13 inputs. We apply the network separately on the left and right eye-candidate face regions. For each pixel in these regions, the 13 NN inputs are the luminance Y, the Cr & Cb chrominance values, and the 10 most important DCT coefficients (with zigzag selection) of the neighboring 8 × 8 pixel area. Using alternative input color spaces such as Lab, RGB or HSV to train the network has not changed its distinction efficiency. The MLP has two outputs, one for each class, namely, eye and noneye, and it has been trained with more than 100 hand-made eye masks that depict eye and noneye area in random frames from the ERMIS [30] database, in images of diverse quality, resolution, and lighting conditions.

The network's output in randomly selected facial images outside the training set is good for locating the eye, as shown in Figure 3(b). However, it cannot provide exact outliers, that is, point locations at the eye boundaries; estimation of *feature points (FP)* is further analyzed in the next section.

To increase speed and reduce memory requirements, the eyes are not detected on every frame using the neural network. Instead, after the eyes are located in the first frame, two square grayscale eye templates are created, containing each of



(a) (b)

Figure 3: (a) Left eye input image (b) network output on left eye, darker pixels correspond to higher output.

the eyes and a small area around them. The size of the templates is half the eye-center distance (bipupil breadth, D_{bp}). For the following frames, the eyes are located inside the two eye-candidate areas, using template matching which is performed by finding the location where the sum of absolute differences (SAD) is minimized.

After head pose is computed, the head is rotated to an upright position and new feature-candidate segmentation is performed on the head using the same rules shown in Table 1, so as to ensure facial features reside inside their respective candidate regions. These regions containing the facial features are used as input for the facial feature extraction stage, described in the following section.

4. AUTOMATIC FACIAL FEATURE DETECTION AND BOUNDARY EXTRACTION

To be able to compute MPEG-4 FAPs, precise feature boundaries for the eyes, eyebrows, and mouth have to be extracted. Eye boundary detection is usually performed by detecting the special color characteristics of the eye area [28], by using luminance projections, reverse skin probabilities, or eye model fitting [17, 41]. Mouth boundary detection in the case of a closed mouth is a relatively easily accomplished task [40]. In case of an open mouth, several methods have been proposed which make use of intensity [17, 41] or color information [18, 28, 42, 43]. Color estimation is very sensitive to environmental conditions, such as lighting or capturing camera's characteristics and precision. Model fitting usually depends on ellipse or circle fitting, using Hough-like voting or corner detection [44]. Those techniques while providing accurate results in high-resolution images are unable to perform well with low video resolution which lack high-frequency properties; such properties which are essential for efficient corner detection and feature border trackability [4] are usually lost due to analogue video media transcoding or low-quality digital video compression.

In this work, nose detection and eyebrow mask extraction are performed in a single stage, while for eyes and mouth which are more difficult to handle, multiple (four in our case) masks are created taking advantage of our knowledge about different properties of the feature area; the latter are then combined to provide the final estimates as shown in Figure 1. Tables 2 and 5 summarize extracted eye and mouth mask notation, respectively, while providing a short qualitative description. In the following, we use the notation $\mathbf{M}_k^x$ to denote the binary mask k of facial feature x, where x is e for eyes, m for mouth, n for nose, and b for eyebrows, and $\mathbf{L}^x$ denotes the respective luminance masks. Additionally, feature size and position validation depends on several relaxed anthropometric constraints; these include t_{asf}^m, t_c^e, t_1^b, t_2^b, t_{b1}^m, t_{c2}^m, t_{b2}^m, t_2^n, t_3^n, t_4^n defined in Table 3, while other thresholds defined in text are summarized in Table 4.

4.1. Eye boundary detection

4.1.1. Luminance and color information fusion mask

This step tries to refine eye boundaries extracted by the neural network described in Section 3 and denoted as ($\mathbf{M}_{\text{nn}}^e$), building on the fact that eyelids usually appear darker than skin due to eyelashes and are almost always adjacent to the iris.

At first, luminance information inside the area depicted by a dilated version of $\mathbf{M}_{\text{nn}}^e$ is used to find a luminance threshold t_b^e:

$$t_b^e = \frac{1}{3}\left(2\left\langle f_c\left(\mathbf{L}^e, \mathbf{M}_{\text{nn}}^e\right)\right\rangle + \min\left(\mathbf{L}^e\right)\right), \tag{1}$$

$$f_c(\mathbf{A}, \mathbf{B}) = \{c_{ij}\}, \quad c_{ij} = \begin{cases} a_{ij}, & b_{ij} \neq 0, \\ 0, & b_{ij} = 0, \end{cases} \tag{2}$$

where $\mathbf{L}^e$ is the luminance channel of the eye-candidate area and $\langle \bullet \rangle$ denotes the average over an image area, and $\min(X)$ denotes the minimum value of area X.

When threshold t_b^e is applied to $\mathbf{L}^e$, a new mask is derived, denoted as $\mathbf{M}_{\text{npp}}^e$. This map includes dark objects near the eye centre, namely, the eyelashes and the iris. From the connected components in $\mathbf{M}_{\text{npp}}^e$ we can robustly locate the one including the iris by estimating its thickness. In particular, we apply a distance transform using the Euclidean distance metric and select the connected component where distance transform obtains its maximum value $\text{DT}_{\max}$, to produce $\mathbf{M}_1^e$ mask as illustrated in Figure 4. The latter includes the iris and adjacent eyelashes. The point where the distance transform equals to $\text{DT}_{\max}$ accurately computes the iris centre.

4.1.2. Edge-based mask

This second approach is based on eyelid detection. Eyelids reside above and below the eye centre, which has already been estimated by the neural network. Taking advantage of their mainly horizontal orientation, eyelids are easily located through edge detection.

We use the canny edge detector [45] mainly because of its good localization performance and its ability to minimize multiple responses to a single edge. Since the canny operator follows local maxima, it usually produces closed curves. Those curves are broken apart into horizontal parts by morphological opening using a 3×1 structuring element; let us denote the result as $\mathbf{M}_{b_1}^e$. Since morphological opening can break edge continuity, we enrich this edge mask by performing edge detection, using a modified canny edge detector. The

Table 2: Summary of eye masks.

Described in	Detects	Depends on	Results
Section 4.1.1	Iris and surrounding dark areas including eyelashes	$\mathbf{L}^e, \mathbf{M}^e_{\text{nn}}$	$\mathbf{M}^e_1$
Section 4.1.2	Horizontal edges produced by eyelids, residing above and below eye centre	$\mathbf{L}^e$, eye centre	$\mathbf{M}^e_2$
Section 4.1.3	Areas of high texture around the iris	$\mathbf{L}^e$	$\mathbf{M}^e_3$
Section 4.1.4	Area with similar luminance to eye area defined by mask $\mathbf{M}^e_{\text{nn}}$	$\mathbf{L}^e, \mathbf{M}^e_{\text{nn}}$	$\mathbf{M}^e_4$

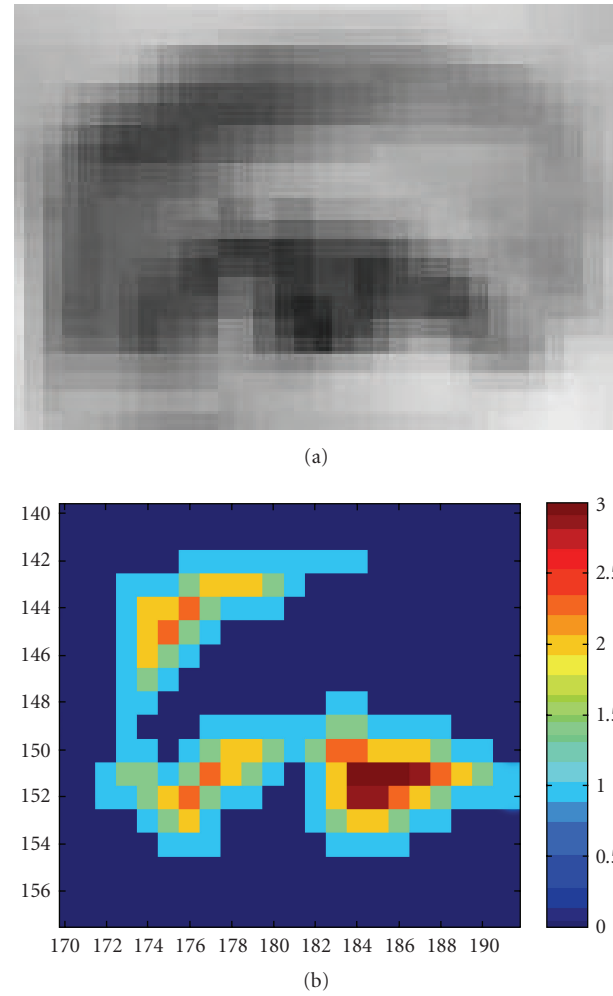
Table 3: Relational anthropometric constraints.

Variable	Value	Refers to
t^m_{asf}	1%	W_f
t^e_c	5%	W_f
t^b_2	5%	D_{bp}
t^n_2	10%	D_{bp}
t^m_{b1}	10%	I_w
t^n_4	15%	D_{bp}
t^m_{c2}	25%	D_{bp}
t^n_3	20%	D_{bp}
t^b_1	30%	D_{bp}
t^m_{b2}	50%	I_w

Table 4: Adaptive thresholds.

Variable	Value	Refers to
t^e_b	$\frac{1}{3}\left(2\langle f_c(\mathbf{L}_e, \mathbf{M}^e_{\text{nn}})\rangle + \min(\mathbf{L}_e)\right)$	L
t^b_E	$\langle \mathbf{M}^b_{E_1}\rangle + \sqrt{\langle (\mathbf{M}^b_{E_1})^2\rangle - \langle \mathbf{M}^b_{E_1}\rangle^2}$	L
t^m_{c1}	$\langle \mathbf{L}^{\text{asfr}}_m\rangle - \sqrt{\langle [\mathbf{L}^{\text{asfr}}_m]^2\rangle - \langle \mathbf{L}^{\text{asfr}}_m\rangle^2}$	L
t^m_1	$\frac{1}{3}\left(2\overline{\mathbf{L}_m} + \min(\mathbf{L}_m)\right)$	L
t^n_1	$\frac{1}{3}\left(\overline{\mathbf{L}_n} + 2\min(\mathbf{L}_n)\right)$	L
t^m_2	90%	NN output
t^e_d	90%	L
	Thresholds	
Variable	Value	
t_σ	10^{-3}	
t_r	128	
t_{vd}	0.8	

L_x: Luminance image of feature x.



(a)

(b)

Figure 4: (a) Left eye input image (cropped). (b) Left eye mask $\mathbf{M}^e_1$ depicting distance transform values of selected object.

latter looks for gradient continuity only in the vertical direction, thus following half of the possible operator movements. Since edge direction is perpendicular to the gradient, this modified canny operator produces mainly horizontal edge lines, resulting in a mask denoted as $\mathbf{M}^e_{b_2}$.

The binary maps $\mathbf{M}^e_{b_1}$ and $\mathbf{M}^e_{b_2}$ are then combined,

$$\mathbf{M}^e_{b_3} = \mathbf{M}^e_{b_1} + \mathbf{M}^e_{b_2}, \tag{3}$$

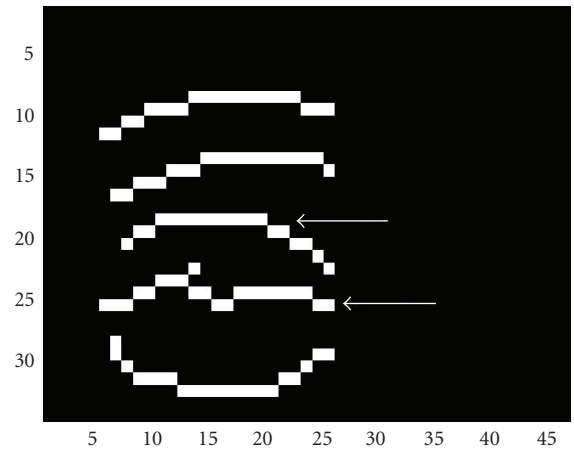
to produce map $\mathbf{M}^e_{b_3}$ illustrated in Figure 5(a). Edges directly above and below the eye centre in map $\mathbf{M}^e_{b_3}$, which are depicted by arrows in Figure 5(a), are selected as eyelids and the space between them as $\mathbf{M}^e_2$, as shown in Figure 5(b).

4.1.3. *Standard-deviation-based mask*

A third mask is created for each of the eyes to strengthen the final mask fusion stage. This mask is created using a region growing technique; the latter usually gives very good segmentation results corresponding well to the observed edges. Construction of this mask relies on the fact that facial texture is more complex and darker inside the eye area and especially in the eyelid-sclera-iris borders than in the areas around them. Instead of using an edge density criterion, we developed a simple but effective new method to estimate both the eye centre and eye mask.

Table 5: Summary of mouth masks.

Described in	Detects	Depends on	Results
Section 4.4.1	Lips and mouth with similar properties to ones trained from the neutral frame	$\mathbf{M}_t^m$, Mouth-candidate image (color)	$\mathbf{M}_1^m$
Section 4.4.2	Horizontal edges caused by lips	$\mathbf{L}^m$	$\mathbf{M}_2^m$
Section 4.4.3	Mouth horizontal extent through lip corner detection. Mouth opening through lip edge detection	$\mathbf{L}^m$	$\mathbf{M}_3^m$



(a)

(b)

Figure 5: (a) Modified canny result. (b) Detected mask $\mathbf{M}_2^e$.

We first calculate the standard deviation of the luminance channel $\mathbf{L}^e$ in $n \times n$ sliding blocks resulting in $\mathbf{I}^e_{\mathrm{std}_n}$. $\mathbf{I}^e_{\mathrm{std}_n}$ is iteratively thresholded with $(1/d)\mathbf{L}^e$, where d is a divisor increasing in each iteration, resulting in $\mathbf{M}^e_{s_{n,d}}$. While d increases, areas in $\mathbf{M}^e_{s_{n,d}}$ dilate, tending to connect with each other.

This operation is performed at first for $n = 3$. The eye centre is selected on the first iteration as the centre of the largest component; for iteration i, the estimated eye centre is denoted as $\mathbf{c}_i$ and the procedure continues while $\|\mathbf{c}_1 - \mathbf{c}_i\| \leq W_f t_c^e$ resulting in binary map $\mathbf{M}^e_{s_{3,f}}$, as illustrated in Figure 6(a). This is an indication that eye area has exceeded its actual borders and is now connected to other subfeatures. The same process is repeated with $n = 6$ resulting in map $\mathbf{M}^e_{s_{6,f}}$ illustrated in Figure 6(b). Different block sizes are used to raise the procedure's robustness to variations of image resolution and eye detail information. Smaller block sizes converge slower to their final map but the combination of both type of maps results in map $\mathbf{M}_3^e$, as in the case of Figure 6(c), ensuring a better result in case of outliers. Examples of outliers include compression artifacts, which induce abrupt illumination variations. For pixel coordinates (i, j), the above are implemented as follows:

$$\begin{aligned}
\mathbf{L}^e &= \{l_{i,j}\},\\
\mathbf{I}^e_{\mathrm{std}_n} &= \{i_{n,i,j}\}, \quad i_{n,i,j} = \sqrt{\langle l_{i,j}^2 \rangle - \langle l_{i,j} \rangle^2},\\
m_{n,d,i,j} &= \begin{cases} 1, & \dfrac{l_{i,j}}{d} > i_{n,i,j}, \\ 0, & \dfrac{l_{i,j}}{d} < i_{n,i,j}, \end{cases} \qquad n = 3, 6,\\
\mathbf{M}^e_{s_{n,d}} &= \{m_{n,d,i,j}\},
\end{aligned} \tag{4}$$

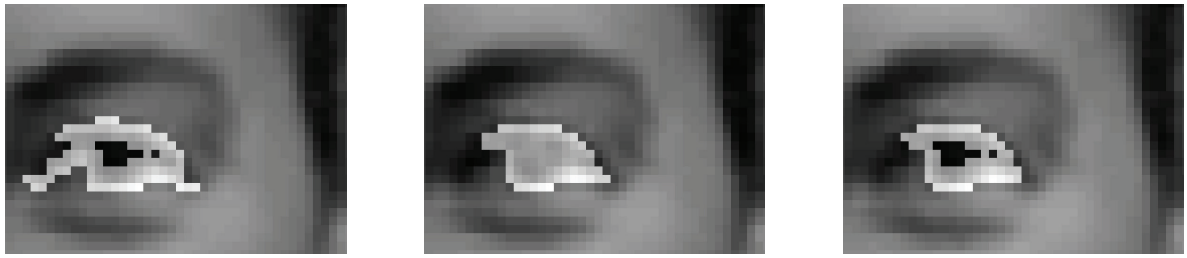
where $d \in (0, \max(\mathbf{L}^e)]$ and $\langle \bullet \rangle$ denotes the mean in the $n \times n$ area surrounding (i, j),

$$\begin{aligned}
f_a(\mathbf{A}, \mathbf{B}) &= \{c_{ij}\}, \quad c_{ij} = a_{ij} b_{ij},\\
\mathbf{M}_3^e &= f_a(\mathbf{M}^e_{s_{2n,f}}, \mathbf{M}^e_{s_{n,f}}).
\end{aligned} \tag{5}$$

The above process is similar to a morphological bottom hat operation with the difference that the latter is rather sensitive to the structuring element size.

4.1.4. Luminance mask

Finally, a second luminance-based mask is constructed for eye/eyelid border extraction. In this mask, we compute the normal luminance probability of $\mathbf{L}^e$ resembling to the mean luminance value of eye area defined by the NN mask $\mathbf{M}^e_{\mathrm{nn}}$. From the resulting probability mask, the areas with a confidence interval of t_d^e are selected and small gaps are closed with morphological filtering. The result is usually a blob depicting the boundaries of the eye. In some cases, the luminance values around the eye are very low due to shadows from the eyebrows and the upper part of the nose. To improve the outcome in such cases, the detected blob is cut vertically at its thinnest points from both sides of the eye centre; the resulting mask's convex hull is then denoted as $\mathbf{M}_4^e$ and illustrated in Figure 7.



(a) (b) (c)

Figure 6: (a) $\mathbf{M}^e_{s_{3,f}}$ eye mask for $n = 3$. (b) $\mathbf{M}^e_{s_{6,f}}$ eye mask for $n = 6$. (c) $\mathbf{M}^e_3$, combination of (a) and (b).

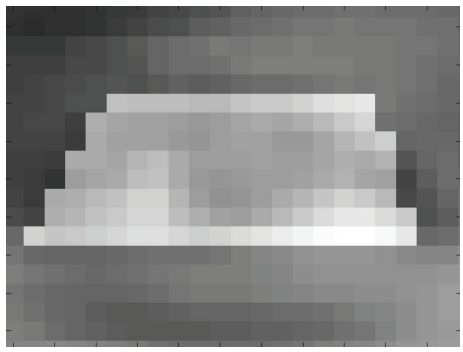
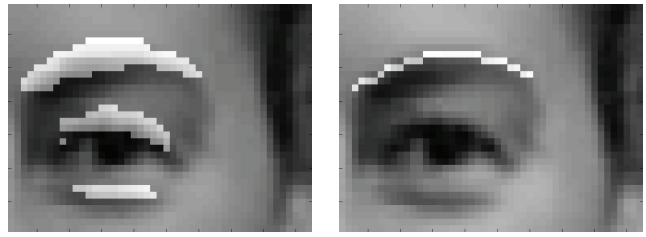


Figure 7: Left eye mask $\mathbf{M}^e_4$.



(a) (b)

Figure 8: (a) Eyebrow candidates. (b) Selected eyebrow mask $\mathbf{M}^b$.

4.2. Eyebrow boundary detection

Eyebrows are extracted based on the fact that they have a simple directional shape and that they are located on the forehead, which due to its protrusion, has a mostly uniform illumination. Each of the left and right eye and eyebrow-candidate images shown in Figure 2 is used for brow mask construction.

The first step in eyebrow detection is the construction of an edge map M^b_E of the grayscale eye/eyebrow-candidate image. This map is constructed by subtracting the dilation and erosion of the grayscale image using a line structuring element st^b_2 pixels long and then thresholding the result as shown in Figure 8(a):

$$\begin{aligned} \mathbf{M}^b_{E_1} &= \delta_s(\mathbf{L}^e), -\varepsilon_s(\mathbf{L}^e), \\ t^b_E &= \left(\langle \mathbf{M}^b_{E_1}\rangle + \sqrt{\left\langle (\mathbf{M}^b_{E_1})^2\right\rangle - \langle \mathbf{M}^b_{E_1}\rangle^2}\right), \\ \mathbf{M}^b_E &= \mathbf{M}^b_{E_1} > t^b_E, \end{aligned} \tag{6}$$

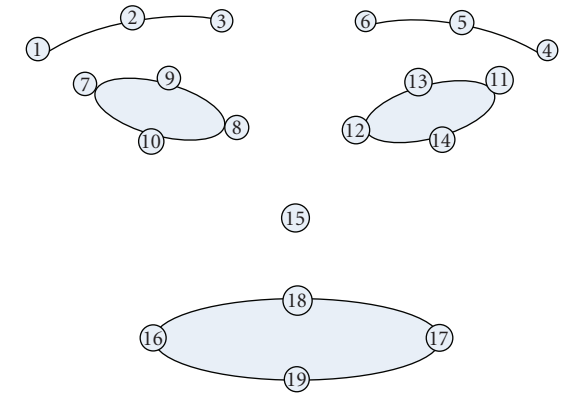
where δ_s, ε_s denote the dilation and erosion operators with structuring element s, and operator ">" denotes the thresholding operator to construct the binary mask $\mathbf{M}^b_E$. The selected edge detection mechanism is appropriate for eyebrows because it can be directional, it preserves the feature's original size and can be combined with a threshold to remove smaller skin anomalies such as wrinkles. The above procedure can be considered as a nonlinear high-pass filter.

Each connected component on the edge map is labeled and then tested against a set of filtering criteria. These criteria were formed through statistical analysis of the eyebrow lengths and positions on 20 persons of the ERMIS database [30]. Firstly, the major axis is found for each component through principal component analysis (PCA). All components whose major axis has an angle of more than 30 degrees with the horizontal plane are removed from the set. From the remaining components, those whose axis length is smaller than t^b_1 are removed. Finally, components with a lateral distance from the eye centre more than $t^b_1/2$ are removed and the top-most remaining is selected resulting in the eyebrow mask $\mathbf{M}^b_{E_2}$. Since eyebrow area is of no importance for FAP calculation, the result can be simplified easily using (7) resulting in $\mathbf{M}^b$ which is depicted in Figure 8(b):

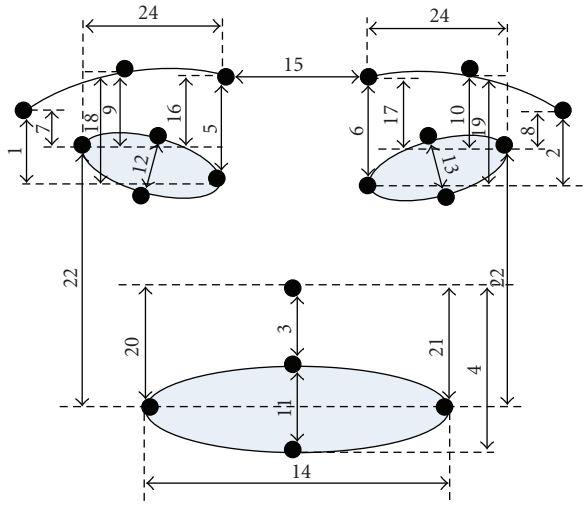
$$\begin{aligned} \mathbf{M}^b &= \{m_{i,j}\}, \\ \mathbf{M}^b_E &= \{m^E_{i,j}\}, \\ m^E_{i,j} &= \begin{cases} 1, & (m_{i,j} = 1) \wedge (m_{i,j'} \neq 1),\ j' < j, \\ 0 & \text{otherwise.} \end{cases} \end{aligned} \tag{7}$$

4.3. Nose localization

The nose is not used for expression estimation by itself, but is a fixed point that facilitates distance measurements for FAP estimation (Figure 9(a)), thus, its boundaries do not have to be precisely located. Nose localization is a feature frequently used for face tracking and usually based on nostril localization; nostrils are easily detected based on their low intensity [46].



(a) Feature points in the facial area



(b) Feature point distances

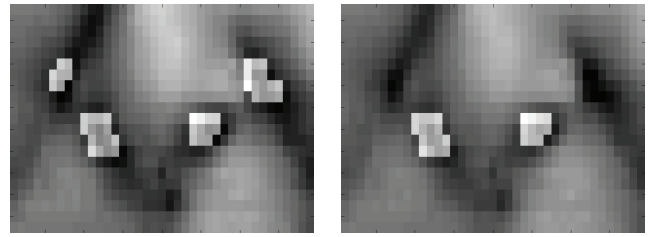
Figure 9

The facial area above the mouth-candidate components area is used for nose location. The respective luminance image is thresholded by t_1^n:

$$t_1^n = \frac{1}{3}\left(\langle \mathbf{L}^n \rangle + 2\min\left(\mathbf{L}^n\right)\right), \quad \mathbf{L}^n : \text{luminance of nose-candidate region.} \tag{8}$$

Connected objects of the derived binary map are labeled. In bad lighting conditions, long shadows may exist along either side of the nose. For this reason, anthropometric data [47] about the distance of left and right eyes (bipupil breadth, D_{bp}) is used to reduce the number of candidate objects: objects shorter than t_2^n and longer than $t_3^n D_{\text{bp}}$ are removed. This has proven to be an effective way to remove most outliers without causing false negative results while generating the nostril mask $\mathbf{M}_1^n$ shown in Figure 10(a).

Horizontal nose coordinate is predicted from the coordinates of the two eyes. On mask $\mathbf{M}_1^n$, each of the connected component horizontal distances from the predicted nose centre is compared to the average internostril distance that is approximately $t_4^n D_{\text{bp}}$ [47], and components with the largest ones are considered as outliers. Those who qualify enter two separate lists, one including left-nostril candidates and one with right-nostril candidates based on their proximity to the left or right eye. Those lists are sorted according to their luminance and the two objects with the lowest values are retained from each list. The largest object is finally kept from each list and labeled as the left and right nostril, respectively, as shown in Figure 10(b). The nose centre is defined as the midpoint of the nostrils.



(a) (b)

Figure 10: (a) Nostril candidates, (b) selected nostrils.

4.4. Mouth detection

4.4.1. Neural network lip and mouth detection mask

At first, mouth boundary extraction is performed on the mouth-candidate facial area depicted in Figure 2. An MLP neural network is trained to identify the mouth region using the neutral image. Since the mouth is closed in the neutral image, a long low-luminance region exists between the lips. The detection of this area, in this work, is carried out as follows.

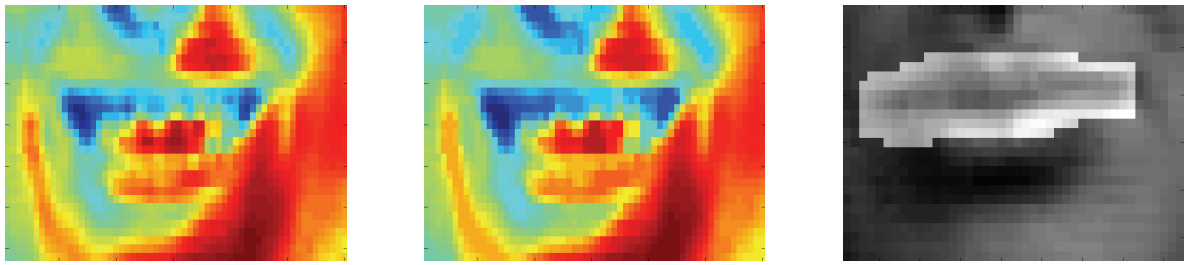
The initial mouth-candidate luminance image $\mathbf{L}^m$ shown in Figure 11(a) is simplified to reduce the presence of noise, remove redundant information, and produce a smooth image that consists mostly of flat and large regions of interest. Alternating sequential filtering by reconstruction (ASFR) (9) is thus performed on $\mathbf{L}^m$ to produce $\mathbf{L}^m_{\text{asfr}}$ shown in Figure 11(b). ASFR ensures preservation of object boundaries through the use of connected operators [48],

$$\begin{aligned} &f_{\text{asfr}}(I) = \beta_n\alpha_n \cdots \beta_2\alpha_2\beta_1\alpha_1(I), \quad n = 1, 2, \ldots, \\ &\alpha_r(I) = \rho^-(f \ominus rB \mid f), \qquad \beta_r(I) = \rho^+(f \oplus rB \mid f), \\ &\qquad\qquad\qquad\qquad\qquad\qquad r = 1, 2, \ldots, \\ &\rho^{+(-)}(g \mid f): \text{ reconstruction closing (opening)} \\ &\qquad\qquad \text{of } f \text{ by marker } g, \end{aligned} \tag{9}$$

where the operations $\oplus$ and $\ominus$ denote the Minkowski dilation and erosion.

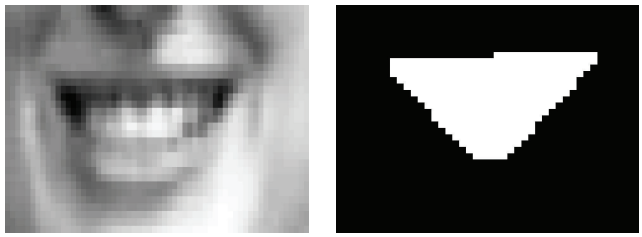
To avoid over simplification, the ASFR filter is applied with a scale of $n \le d_m^w \cdot t_{\text{asf}}^m$, where d_m^w is the width of $\mathbf{L}^m$. The luminance image is then thresholded by t_1^m:

$$t_1^m = \frac{1}{3}\left(2\overline{\mathbf{L}^m} + \min\left(\mathbf{L}^m_{\text{asfr}}\right)\right), \tag{10}$$



(a) (b) (c)

Figure 11: Extraction of training image: (a) initial luminance map $\mathbf{L}^m$, (b) filtered image $\mathbf{L}^m_{\text{asfr}}$, (c) extracted mask $\mathbf{M}^m_{t_1}$.



(a) (b)

Figure 12: (a) Luminance image, (b) NN mouth mask $\mathbf{M}^m_1$.

(a) (b)

Figure 13: (a) Initial binary edge map. (b) Output mask $\mathbf{M}^m_{b_2}$.

and connected objects on the resulting binary mask $\mathbf{M}^m_{t_1}$ are labeled as shown in Figure 11(c).

The major axis of each connected component is computed through PCA analysis, and the one with the longest axis is selected. The latter is subsequently dilated vertically and the resulting mask $\mathbf{M}^m_t$ is produced, which includes the lips. Mask $\mathbf{M}^m_t$ shown in Figure 11(c) is used to train a neural network to classify the mouth and nonmouth areas accordingly. The image area included by the mask corresponds to the mouth class and the image outside the mask to the nonmouth one. The perceptron has 13 inputs and its architecture is similar to that of the network used for eye detection.

The neural network trained on the neutral-expression frame is then used on other frames to produce an estimate of the mouth area: neural network output on the mouth-candidate image is thresholded by t^m_2 and those areas with high confidence are kept to form a binary map containing several small subareas. The convex hull of these areas is calculated to generate mask $\mathbf{M}^m_1$ as shown in Figure 12.

4.4.2. Generic edge connection mask

In this second approach, the mouth luminance channel is again filtered using ASFR for image simplification. The horizontal morphological gradient of $\mathbf{L}^m$ is then calculated similarly to the eyebrow binary edge map detection resulting in $\mathbf{M}^m_{b_1}$ shown in Figure 13(a). Since the nose has already been detected, its vertical position is known. The connected elements of $\mathbf{M}^m_{b_1}$ are labeled and those too close to the nose are removed. From the rest of the map, very small objects (less than $t^m_{b1} I_w$, where I_w is the map's width) are removed.

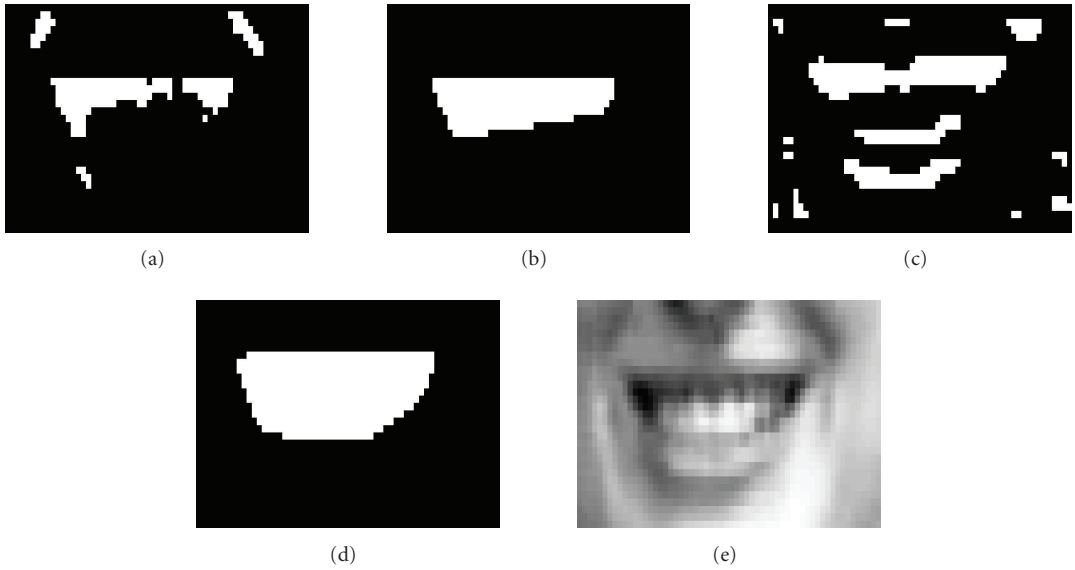
Figure 14: Mouth-candidate area depicting nonuniform illumination.

Morphological closing is then performed so that those whose distance is less than $t^m_{b2} I_w$ connect together, in order to obtain mask $\mathbf{M}^m_{b_2}$ as shown in Figure 13(b). The longest of the remaining objects in horizontal sense is selected as mouth mask $\mathbf{M}^m_2$.

4.4.3. Lip-corner luminance and edge information fusion mask

The problem of most intensity-based methods that try to estimate mouth opening is the visibility of upper teeth, especially if they appear between the upper and lower lip altering saturation and intensity uniformity as illustrated in Figure 14.

A new method is proposed next to cope with this problem. First, the mouth-candidate luminance channel $\mathbf{L}^m$ is



(a) (b) (c) (d) (e)

Figure 15: (a) Mask $\mathbf{M}_{c_1}^m$ with removed background outliers, (b) mask $\mathbf{M}_{c_2}^m$ with apparent teeth, (c) horizontal edge mask $\mathbf{M}_{c_3}^m$, (d) output mask $\mathbf{M}_3^m$, (e) input image.

thresholded using a low threshold t_{c1}^m providing an estimate of the mouth interior area, or the area between the lips in case of a closed mouth. The threshold used is estimated adaptively:

$$\begin{aligned} \mathbf{M}_{c_{1a}}^m &= \mathbf{L}_{\text{asfr}}^m < t_{c1}^m, \\ t_{c1}^m &= \left(\left\langle \mathbf{L}_{\text{asfr}}^m \right\rangle - \sqrt{\left\langle \left[\mathbf{L}_{\text{asfr}}^m \right]^2 \right\rangle - \left\langle \mathbf{L}_{\text{asfr}}^m \right\rangle^2} \right), \end{aligned} \tag{11}$$

where operator "<" again stands for the thresholding process.

In the resulting binary map, all connected objects adjacent to the border are removed, thus removing facial background outliers, resulting in mask $\mathbf{M}_{c_1}^m$ shown in Figure 15(a). We now examine two cases separately: either we have no apparent teeth and the mouth area is denoted by a cohesive dark area (case 1) or teeth are apparent and thus two dark areas appear at both sides of the teeth (case 2). It should be noted that those areas appear even in large extensive smiles. The largest connected object is then selected from $\mathbf{M}_{c_1}^m$ and its centroid is found. If the horizontal position of its centroid is near the horizontal nose position, case 1 is selected, otherwise case 2 is assumed to occur and two dark areas appear at both sides of the teeth. To assess horizontal noise centre proximity, we use a distance threshold of $t_{c2}^m D_{\text{bp}}$. The two cases are quite distinguishable through this process. In case 2, the second largest connected object is also selected. A new binary map is created containing either one object in case 1 or both objects in case 2; the convex hull of this map is then calculated and mask $\mathbf{M}_{c_2}^m$ is produced, depicted in Figure 15(b).

The detected lip corners provide a robust estimation of mouth horizontal extent but are not adequate to detect mouth opening. Therefore, mask $\mathbf{M}_{c_2}^m$ is expanded to include the lower lips. An edge map is created as follows: the mouth image gradient is calculated in the horizontal direction, and is thresholded by the median of its positive values, as shown in Figure 15(c). This mask, denoted as $\mathbf{M}_{c_3}^m$, contains objects close to the lower middle part of the mouth, which are sometimes missed because of the lower teeth. The two masks, $\mathbf{M}_{c_2}^m$ and $\mathbf{M}_{c_3}^m$, have to be combined to a final mask. An effective way of achieving this is to keep from both masks objects which are close to each other. Since $\mathbf{M}_{c_2}^m$ may contain objects belonging to lower parts of the mouth area, it is expanded downwards by dilation with a nonsymmetric vertical structuring element, resulting in mask $\mathbf{M}_{c_{2d}}^m$. Morphological reconstruction [49] is then used to combine the masks together by using the area belonging to both $\mathbf{M}_{c_3}^m$ and $\mathbf{M}_{c_{2d}}^m$ as input and objects belonging to either mask (12) as marker. Final mask $\mathbf{M}_3^m$ is shown in Figure 15(d),

$$\begin{aligned} \mathbf{M}_{c_{23}}^m &= f_a(\mathbf{M}_{c_3}^m, \mathbf{M}_{c_{2d}}^m), \\ f_a(\mathbf{A}, \mathbf{B}) &= \{c_{ij}\}, \quad c_{ij} = \{a_{ij} b_{ij}\}, \\ f_o(\mathbf{A}, \mathbf{B}) &= \{c_{ij}\}, \quad c_{ij} = \begin{Bmatrix} a_{ij}, b_{ij} = 0 \\ b_{ij}, a_{ij} = 0 \end{Bmatrix}, \\ \mathbf{M}_3^m &= \rho(\mathbf{M}_{c_{23}}^m, f_o(\mathbf{M}_{c_{2d}}^m, \mathbf{M}_{c_3}^m)), \end{aligned} \tag{12}$$

where $\rho(\mathbf{B}, \mathbf{A})$ denotes the reconstruction of $\mathbf{A}$ with marker $\mathbf{B}$.

5. FINAL MASKS GENERATION AND CONFIDENCE ESTIMATION

Each facial feature's masks must be fused together to produce a final mask for that feature. The most common problems, especially encountered in low quality input images, include connection with other feature boundaries or mask dislocation due to noise, as depicted in Figure 16. In some cases, some masks may have completely missed their goal and provide a completely invalid result. Outliers such as illumination

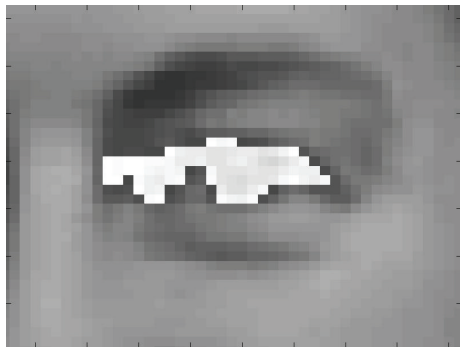


Figure 16: Noisy color and edge information cause problems in the extraction of this mask.

changes and compression artifacts cannot be predicted and so individual masks have to be re-evaluated and combined on each new frame.

5.1. Validation of eye and mouth masks

The proposed algorithms presented in Section 4 produce a mask $\mathbf{M}^b$ for each eyebrow, nose coordinates, four intermediate mask estimates $\mathbf{M}^e_{1\ldots4}$ for each eye and three intermediate mouth mask estimates $\mathbf{M}^m_{1\ldots3}$. The four masks for each eye and three mouth masks must be fused to produce a final mask for each feature. Since validation can only be done on the end result of each intermediate mask, we unfortunately cannot give different parts of each intermediate mask different confidence values, so each pixel of those masks will share the same value. We propose validation through testing against a set of anthropometric conformity criteria. Since, however, some of these criteria relate either to aesthetics or to transient feature properties, we cannot apply strict anthropometric judgment.

For each mask k of feature x, we employ a set of validation measurements $V^x_{k,i}$, denoted by i, which are then combined to a final validation tag $V^x_{k,f}$ for that mask. Each measurement produces a validation estimate value depending on how close it is to the usually expected feature shape and position, in the neutral expression. Expected values for these measurements are defined from anthropometry data [33] and from images extracted from video sequences of 20 persons in our database [30]. Thus, a validation tag between [0,1] is attached to each mask, with higher values denoting proximity to the most expected measurement values.

All validation measurements are based on distances defined in Table 6. Given these definitions, eye mask validation is based on four tags specified in Table 7, concerning individual eye dimensions, relations between the two eyes and relations between each eye and the corresponding eyebrow. Finally, mouth map validation is based on four tags referring to distance measurements specified in Table 8. In the following, validation value of measurement i for mask k of feature x will be denoted as $V^x_{k,i} \in [0,1]$ where $V^x_{k,i}$ is forced into [0,1], that is, if $V^x_{k,i} > 1$, then $V^x_{k,i} = 1$ and if $V^x_{k,i} < 0$, then $V^x_{k,i} = 0$.

We want masks with very low validation tags to be discarded from the fusion process and thus those are also prevented from contribution on final validation tags; therefore, we ignore those with $V^x_{k,f} < (t_{vd} \cdot \langle V^x_{k,i} \rangle_i)$. Final validation tag for mask k is then calculated as follows:

$$V^x_{k,f} = \langle V^x_{k,i'} \rangle_{i'}, \quad i' : V^x_{k,i'} \geq t_{vd} \langle V^x_{k,i} \rangle_i,\ i \in \mathbb{N}_n. \tag{13}$$

Table 6: Mask validation distances.

d_1	Distance of eye's top horizontal coordinate and eyebrow's middle bottom horizontal coordinate
d_2	Eye width
d_3	Eye height
d_4	Distance of eye's middle vertical coordinate and eyebrow's middle vertical coordinate
d_5	Eyebrow width
d_6	D_{bp}, bipupil breadth
d_7	Distance of eye's middle vertical coordinate from mouth's middle vertical coordinate
d_8	Mouth width
d_9	Mouth height
d_{10}	Sellion-Stomion length
d_{11}	Sellion-Subnasion length

5.2. Mask fusion

Each of the intermediate masks represents the best-effort result of the corresponding mask-extraction method used. Multiple eye and mouth masks must be merged to produce final mask estimates for each feature. The mask fusion method is based on the assumption that having multiple masks for each feature lowers the probability that all of them are invalid since each of them produces different error patterns. It has been proven in committee machine (CM) theory [50, 51] that for the desired output t the combination error $y_{\text{comb}} - t$ from different machines f_i is guaranteed to be lower than the average error:

$$y_{\text{comb}} = \frac{1}{M} \sum y_i,$$
$$(y_{\text{comb}} - t)^2 = \frac{1}{M} \sum_i (y_i - t)^2 - \frac{1}{M} \sum_i (y_i - y_{\text{comb}})^2. \tag{14}$$

Since intermediate masks have a validation tag which represents their "plausibility" of being actual masks for the feature they represent, it seems natural to combine them by giving more credit to those which have a higher validation value on one hand, and on the other to ignore those that we are sure will not contribute positively on the result. Furthermore, according to the specific qualities of each input, we would like to favor specific masks that are known to perform better on those inputs, that is, give more trust to color-based extractors when it is known that input has good color quality, or to the neural network-based masks when the face resolution is enough for the network to perform adequate border detection.

Regarding input quality, two parameters can be taken into account: image resolution and color quality; since

Table 7: Anthropometric validation measurements used for eye masks. Note that (eye width)/(bipupil breadth) = 0.49 [33].

Validation tag	Measurement	Description
$V^e_{k,1}$	$1 - \lvert (d_1)/(d_6/4) - 1 \rvert$	Distance of the eye's topmost centre from the corresponding eyebrow's bottom centre.
$V^e_{k,2}$	$1 - \lvert 1 - (d_2/d_6)/0.49 \rvert$	Eye width compared to left & right eye distance.
$V^e_{k,3}$	$0.3 - (d_3 - d_2)/d_2$	Relation of eye width and height
$V^e_{k,4}$	$1 - \lvert d_4 \rvert / d_5$	Horizontal alignment of the eye and respective eyebrow

Table 8: Anthropometric validation measurements used for mouth masks. Note that (bichelion breadth)/(bipupil breadth) = 0.82 and (stomion-subnasion length)/(bipupil breadth) = 0.344 [33].

Validation tag	Measurement	Description
$V^m_{k,1}$	$1 - \lvert d_7 \rvert / d_6$	Horizontal mouth centre, in comparison with the inter-eye centre coordinate.
$V^m_{k,2}$	$1 - \left\lvert \frac{d_8}{d_6} \frac{1}{0.82} - 1 \right\rvert$	Mouth width in comparison with bipupil breadth
$V^m_{k,3}$	1 if $d_9 < (1.3 d_6)$ else $d_9/(1.3 d_6)$	Mouth height in comparison with bipupil breadth
$V^m_{k,4}$	$1 - \left\lvert 1 - \frac{(d_{10} - d_{11})}{d_6} \frac{1}{0.344} \right\rvert$	Nose distance from top lip

nonsynthetic training data for the latter is difficult to acquire, we have found that a good estimator can be the chromatic deviation measured on the face skin area: very large variability in chromatic components is a good indicator for color noise presence. Therefore, σ_{Cr}, σ_{Cb} are less than t_σ for good color quality and much larger for poor quality images. Regarding resolution, we have found that the proposed neural-network-based detector performs very well in sequences where $D_{\mathrm{bp}} > t_r$ pixels, where D_{bp} denotes the bipupil breadth.

In the following, we use the following notation: final masks for left eye, right eye, and mouth are denoted as before as $\mathbf{M}^{e_L}_f$, $\mathbf{M}^{e_R}_f$, $\mathbf{M}^m_f$. For intermediate mask k of feature x, variable $V^x_{k,f}$ determines which masks are favored according to their final validation values and variable g^k determines which masks extractors are favored according to input characteristics. Moreover, each pixel-element on the final mask $\mathbf{M}^x_f$ is denoted as m^x_f and each pixel-element on the kth intermediate mask $\mathbf{M}^x_k$ as m^x_k, $k \in \mathbb{N}_n$, where pixel coordinates are omitted for clarity. Moreover, since we would like masks to be fused in a per-pixel basis, not all pixels on an output mask will necessarily derive from the same intermediate masks. Therefore, each pixel on the output mask will have a validation value v^x_f which will reflect mask validation and extractor suitability of the masks it derived; values of v^x_f for all pixels form validation values of final mask, V^x_f.

Let us denote the function between $m^x_f \in \{0,1\}$, $v^x_f \in [0,1]$, and $m^x_k \in \{0,1\}$ as

$$\begin{aligned} v^x_f &= f\left(m^x_k; V^x_{k,f}, g^k\right), \\ m^x_f &= F\left(v^x_f\right), \end{aligned} \tag{15}$$

then our requirements can be expressed as follows.

(1) If all masks k agree that a pixel m^x_k does not belong to the feature x, then this should be reflected on the fusion result regardless of validation tags $V^x_{k,f}$:

$$\text{if } \forall k \in \mathbb{N}_n, \quad m^x_k = 0 \Longrightarrow m^x_f = 0. \tag{16}$$

(2) We require that gating variable g^k should be balanced according to the number of masks:

$$\sum_{k=1}^{n} g^k = n. \tag{17}$$

(3) If all masks k agree that a pixel m^x_f does belong to feature x with maximum confidence, then this should be reflected on the fusion result:

$$\text{if } \forall k \in \mathbb{N}_n, \quad m^x_k = 1 \wedge V^x_{k,f} = 1 \Longrightarrow m^x_f = 1,\ v^x_f = 1. \tag{18}$$

(4) If all masks k have failed, then no mask should be created as a fusion result:

$$\forall k \in \mathbb{N}_n, \quad V^x_{k,f} = 0 \Longrightarrow m^x_f = 0. \tag{19}$$

(5) If one mask has failed, then the result should depend only on remaining masks:

$$\exists k_0 \in \mathbb{N}_n : V^x_{k_0,f} = 0 \Longrightarrow m^x_f = \underset{k \in \mathbb{N}_n - \{k_0\}}{f} \left(m^x_k; V^x_{k,f}, g^k\right). \tag{20}$$

(6) Fusion with a better input mask should produce a higher value on the output for the pixels deriving from this mask:

$$\begin{aligned} &\text{if } V^{x_1}_{k_0,f} > V^{x_2}_{k_0,f}, \quad \forall k \in \mathbb{N}_n - \{k_0\}, \\ &\text{it is } V^{x_1}_{k,f} = V^{x_2}_{k,f} \text{ and the same holds for all} \\ &m^{x_j}_k \neq 0, \quad g^{k,j}, \quad j = 1,2 \text{ then } v^{x_2}_f > v^{x_2}_f. \end{aligned} \tag{21}$$

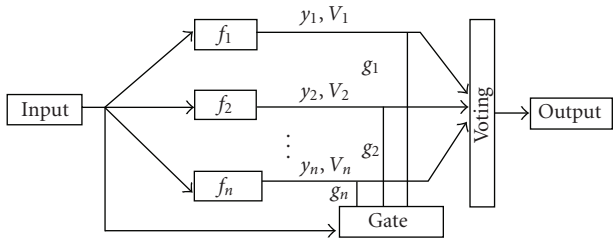


Figure 17: The dynamic committee machine model.

(7) If an input mask derives from a more trusted mask extractor, then pixels deriving from this mask should be associated with a higher value:

$$\text{if } g^{k,1} > g^{k,2}, \quad \forall k \in \mathbb{N}_n - \{k_0\} \text{ it is } V_{k,f}^{x_1} = V_{k,f}^{x_2},$$
$$\text{and the same holds for all } m_k^{x_j} \neq 0, \quad V_{k,f}^{x_j}, \quad j = 1,2 \quad (22)$$
$$\text{then } v_f^{x_2} > v_f^{x_2}.$$

To fulfill these requirements in this work, we propose a fusion method based on the idea of dynamic committee machines (DCM) which is depicted in Figure 17. In a static CM, the voting weight for a component is proportional to its error on a validation set. In DCMs, input is directly involved in the combining mechanism through a gating network (GN), which is used to modify those weights dynamically.

The machine's inputs are intermediate masks $\mathbf{M}_k^x$, $V_{k,f}^x$ is considered as the confidence of each input and variable g^k has a "gating" role. Final masks $\mathbf{M}_f^{e_L}$, $\mathbf{M}_f^{e_R}$, $\mathbf{M}_f^m$ are considered as the machine's output.

Each pixel-element m_f^x on the final mask $\mathbf{M}_f^x$ is calculated from the n masks as follows:

$$f\left(m_k^x; V_{k,f}^x, g^k\right) = \frac{1}{n}\sum_{k=1}^{n} m_k^x V_{k,f}^x g^k, \quad (23)$$

$$F\left(v_f^x\right) = \begin{cases} 0, & v_f^x < \left(\left\langle V_f^x \right\rangle \mid v_f^x > 0\right), \\ 1, & \text{otherwise.} \end{cases} \quad (24)$$

The role of gating variable g^k is used to favor color-aware feature extraction methods ($\mathbf{M}_1^e$, $\mathbf{M}_1^m$) in images of high-color quality and resolution; gating variable g^i is defined as follows:

$$g^k = \begin{cases} n - \left(\dfrac{n-1}{n}\right), & k = 1,\ D_{\text{bp}} > t_r,\ \sigma_{\text{Cr}} < t_\sigma,\ \sigma_{\text{Cb}} < t_\sigma, \\ \dfrac{1}{n}, & k \neq 1,\ D_{\text{bp}} > t_r,\ \sigma_{\text{Cr}} < t_\sigma,\ \sigma_{\text{Cb}} < t_\sigma, \\ 1, & \text{otherwise,} \end{cases} \quad (25)$$

where D_{bp} the bipupil width in pixels, σ_{Cr}, σ_{Cb} the standard deviation of the Cr, Cb channels, respectively, inside the facial area. It is not difficult to see that (23)–(25) satisfy (16)–(22).

Tables 9 and 10 illustrate mask fusion examples for the left eye and mouth where some of the masks are problematic. Validation tags refer to the corresponding mask validation tag while D_{bp} is quoted as an indication of the sequence resolution. For illustration purposes, the feature points extracted from the final masks are presented verifying the precise extraction of the features and feature points, based on the mask fusion process.

5.3. Eye, eyebrow, and mouth mask confidence estimation

Confidence values are needed for expression analysis and are thus propagated from mask extraction to the corresponding FPs, FAPs, and the expression evaluation stage. Their role is to indicate the confidence that a given feature has been correctly extracted and therefore the measure by which expression analysis should rely on a specific feature. To estimate confidence, we have used extracted feature resemblance to mean anthropometry data from [33]. Since data for eyebrow sizes was not available in the literature, confidence values were expanded to rely also on information such as facial feature size constancy and face symmetry.

Confidence values can be attached to each final mask and are denoted as $C^e, C^b, C^m \in [0,1]$. Confidence values vary between 0 and 1 with the latter indicating the best case. For the nose, no confidence value is estimated and is always assumed that $C^n = 1$. Those values are generated through a set of criteria, which complement final validation tags V_f used for fusion; these criteria relate to

(1) size constancy over time, producing C_{med}^b, C_{med}^e;
(2) face symmetry, producing C_s^e;
(3) and anthropometric measurement conformance, producing C_1^e, C_2^e, C_1^m.

These values are calculated as follows.

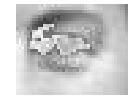
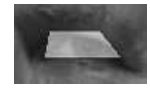
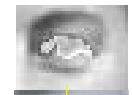
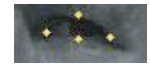
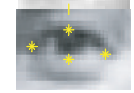
(1) With the exception of mouth, facial feature width is mostly constant even in intense expressions. Measured width for eyebrows w_i^b and each of the eyes $w_i^{e_L}$, $w_i^{e_R}$ is examined in each frame i the median value $\tilde{w}^x$ over the last 10 frame period for feature x is calculated. In each frame, similarity between w_i^x and $\tilde{w}^x$ on the last 10 frames is used as an estimate for C_{med}^b for the eyebrows and C_{med}^e for the eyes:

$$C_{\text{med},i}^x = 1 - \left| w_i^x - \text{med}\left(w_j^x,\ j = i-10 \ldots i\right) \right| \left(w_i^x\right)^{-1}. \quad (26)$$

(2) $C_s^e \in [0,1]$ denotes shape similarity between the left and right upper eyelid; exploiting the symmetry of the face, we estimate the resemblance between the upper parts of left and right eyelids. Let us define $\mathbf{X}^L$, $\mathbf{X}^R$ as matrices containing the horizontal coordinates of the left and right upper eyelid boundaries; a value C_s indicating their similarity can be calculated as a two-dimensional correlation coefficient between the two vectors,

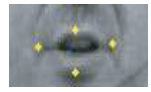
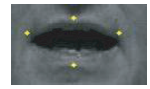
$$C_s^e = \frac{\sum_n \left(\left(\mathbf{X}_n^L - \left\langle \mathbf{X}^L \right\rangle\right)\left(\mathbf{X}_n^R - \left\langle \mathbf{X}^R \right\rangle\right)\right)}{\sqrt{\left(\sum_n \left(\mathbf{X}_n^L - \left\langle \mathbf{X}^L \right\rangle\right)^2\right)\left(\sum_n \left(\mathbf{X}_n^R - \left\langle \mathbf{X}^R \right\rangle\right)^2\right)}}. \quad (27)$$

Table 9: Examples of mask fusion on the left eye with corresponding validation tags and detected feature points.

Sequence-frame	kk-1002	V_f^e	kk-1998	V_f^e	rd-12259	V_f^e	al-27	V_f^e
Mask								
$\mathbf{M}_{\text{nn}}^e$								
$\mathbf{M}_1^e$		0.825		0.813		0.823		0.839
$\mathbf{M}_2^e$		0.782		0.581		0.763		0.810
$\mathbf{M}_3^e$		0.866		0.733		0.716		0.787
$\mathbf{M}_4^e$		0.883		0.917		0.826		0.872
$\mathbf{M}_f^e$								
FPs								
	D_{bp}: 58 px		D_{bp}: 58 px		D_{bp}: 96 px		D_{bp}: 36 px	

D_{bp} denotes bipupil breadth in pixels and is quoted as an image resolution indicative.

Table 10: Examples of mouth mask fusion with corresponding validation tags and detected feature points.

Sequence-frame	kk-1014	V_f^m	rd-1113	V_f^m
Mask				
$\mathbf{M}_1^m$		0.820		0.538
$\mathbf{M}_2^m$		0.868		0.752
$\mathbf{M}_3^m$		0.828		0.821
$\mathbf{M}_f^m$				
FPs				

(3) C^e, C^m, C^b are calculated using measurements based on anthropometry from [33]. Table 11 summarizes estimation of C_1^e, C_2^e, C_1^m.

Confidence values for features are estimated by averaging on the previously defined criteria and final mask validation tags as follows:

$$C^e = \langle V_f^e, C_1^e, C_2^e, C_s^e, C_{\text{med}}^e \rangle,$$
$$C^m = \langle V_f^m, C_1^m \rangle, \tag{28}$$
$$C^b = C_{\text{med}}^b.$$

6. EXPRESSION ANALYSIS

An overview of the expression recognition process is shown in Figure 1. At first, 19 feature points (FPs) are calculated from the corresponding feature masks. Those FPs have to be compared with the FPs of the neutral frame, so as to measure movement and estimate FAPs. FAPs are then used to evaluate expression profiles, providing the recognized expression.

6.1. From masks to feature points

Left-, right-, top-, and bottom-most coordinates of the final masks $\mathbf{M}_f^{e_L}$, $\mathbf{M}_f^{e_R}$, $\mathbf{M}_f^m$, left right and top coordinates of $\mathbf{M}_f^{b_L}$, $\mathbf{M}_f^{b_R}$, as well as nose coordinates, are used to define the 19 feature points (FPs) shown in Table 12, Figures 18 and 9(a). Feature point x is then assigned with confidence C_x^{FP} by inheriting the confidence level (C^e, C^m, C^b, C^n) of the final mask from which it derives.

Table 11: Anthropometric evaluation [33] for eye and mouth location and size.

Description	Confidence measure
Bientocanthus breadth	D_7^a
Biectocanthus breadth	D_5^a
Bicheilion breadth	D_{10}^a
$(D_5^a - D_7^a)/2$	D_{ew}^a
Eye position/eye distance	$C_1^e = 1 - \lvert D_5^{an} - D_5^n \rvert / D_5^{an}$
Eye width	$C_2^e = 1 - \lvert D_{ew}^{an} - D_{ew}^n \rvert / D_{ew}^{an}$
Mouth	$C_1^m = 1 - \lvert D_{10}^{an} - D_{10}^n \rvert / D_{10}^{an}$

$D_i^{an} = D_x^a / D_7^a$: a denotes that distance i derives from [33]; n denotes that value is normalized by division with D_7^a.

Table 12: Feature points.

FP no.	MPEG-4 FP [6]	FP name
01	4.5	Outer point of left eyebrow
02	4.3	Middle point of left eyebrow
03	4.1	Inner point of left eyebrow
04	4.6	Outer point of right eyebrow
05	4.4	Middle point of right eyebrow
06	4.2	Inner point of right eyebrow
07	3.7	Outer point of left eye
08	3.11	Inner point of left eye
09	3.13	Upper point of left eyelid
10	3.9	Lower point of left eyelid
11	3.12	Outer point of right eye
12	3.8	Inner point of right eye
13	3.14	Upper point of right eyelid
14	3.10	Lower point of right eyelid
15	9.15	Nose point
16	8.3	Left corner of mouth
17	8.4	Right corner of mouth
18	8.1	Upper point of mouth
19	8.2	Lower point of mouth

6.2. From FP to FAP estimation

A 25-dimensional distance vector (D_v) is created containing vertical and horizontal distances between 19 extracted FPs, as shown in Figure 9(b). Distances are not measured in pixels, but in normalized scale-invariant MPEG-4 units, that is, ENS, MNS, MW, IRISD, and ES [6]. Unit bases are measured directly from FP distances on the neutral image; for example, ES is calculated as $|FP_9, FP_{13}|$.

The distance vector is created once for the neutral-expression image (D_v^n) and for each of the subsequent frames (D_v). FAPs are calculated by comparing D_v^n and D_v. Each FAP depends on one or more elements of D_v thus some FAPs are over defined; the purpose of calculating a FAP from more distances than necessary is to increase estimation robustness which is accomplished by considering the confidence levels of each distance element. Elements in D_v are calculated by measuring the FP distances illustrated in Figure 9(b). Uncertainty in FP coordinates should reflect to corresponding FAPs; therefore, distances needed to calculate an FAP are weighted according to the confidence of the corresponding FP from which they derive.

A value C_i^{FAP} indicating the confidence of FAP i is estimated as $C_i^{\text{FAP}} = \langle C_{\text{Y}}^{\text{FP}} \rangle$, Y:set of FPs used to estimate FAP i. Correspondences between FAPs and corresponding distance vector elements are illustrated in Table 13.

Table 13: Example of FAPs and related distances.

MPEG4 FAP	Description	Distance number
F_3	open_jaw	11
F_4	lower_top_midlip	3
F_5	raise_bottom_midlip	4
$F_6 + F_7$	widening_mouth	14
$F_{19} + F_{21}$	close_left_eye	12
$F_{20} + F_{22}$	close_right_eye	13
F_{31}	raise_left_inner_eyebrow	5,16
F_{32}	raise_right_inner_eyebrow	6,17
F_{33}	raise_left_medium_eyebrow	18,9
F_{34}	raise_right_medium_eyebrow	19,10
F_{35}	raise_left_outer_eyebrow	7,1
F_{36}	raise_right_outer_eyebrow	8,2
F_{37}	squeeze_left_eyebrow	24
F_{38}	squeeze_right_eyebrow	25
$F_{37} + F_{38}$	squeeze_eyebrows	15
F_{59}	raise_left_outer_cornerlip	22
F_{60}	raise_right_outer_cornerlip	23

6.3. Facial expression recognition and human computer interaction

In our former research on expression recognition, a ru le-based system was created, characterising a user's emotional state in terms of the six universal, or archetypal, expressions (joy, surprise, fear, anger, disgust, sadness). We have created rules in terms of the MPEG-4 FAPs for each of these expressions, by analysing the FAPS extracted from the facial expressions of the Ekman dataset [7]. This dataset contains several images for every one of the six archetypal expressions, which, however, are rather exaggerated. As a result, rules extracted from this dataset do not perform well if used in real human-computer interaction environments. Psychological studies describing the use of quadrants of emotion's wheel (see Figure 19) [52] instead of the six archetypal expressions provide a more appropriate tool in such interactions. Therefore, creation of rules describing the first three quadrants—no emotion is lying in the fourth quadrant—is necessary.

To accomplish this, facial muscle movements were translated into FAPs while each expression's FAPs on every quadrant were experimentally verified through analysis of prototype datasets. Next, the variation range of each FAP was computed by analysing real interactions and corresponding video sequences as well as by animating synthesized exam-

Figure 18: The 19 detected feature points. Automatic head-pose recovery has been performed.

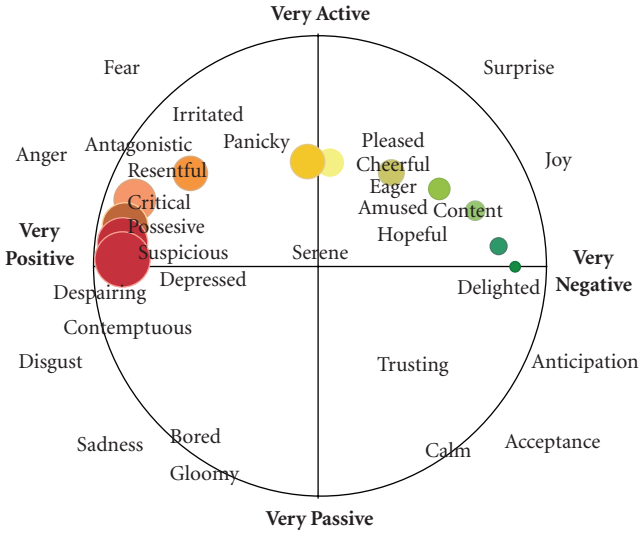


Figure 19: The activation-emotion space.

ples. Table 14 illustrates three examples of rules that were created based on the developed methodology.

In order to use these rules in a system dealing with the continuous activation-emotion space and fuzzy representation, we transformed the rules replacing the range of variation with the terms *high, medium, low* after having normalized the corresponding partitions. The full set of rules can be found in [53].

In the process of exploiting the knowledge contained in the fuzzy rule base and the information extracted from each frame in the form of FAP measurements, with the aim to analyze and classify facial expressions, a series of issues have to be tackled.

(i) FAP activation degrees need to be considered in the estimation of the overall result.

(ii) The case of FAPs that cannot be estimated, or equivalently are estimated with a low degree of confidence, needs to be considered,

$$\text{if } x_1, x_2, \ldots, x_n, \quad \text{then } y. \tag{29}$$

The conventional approach to the evaluation of fuzzy rules of the form described in (29) is as follows [54]:

$$y = t(x_1, x_2, \ldots, x_n), \tag{30}$$

where t is a fuzzy t-norm, such as the minimum

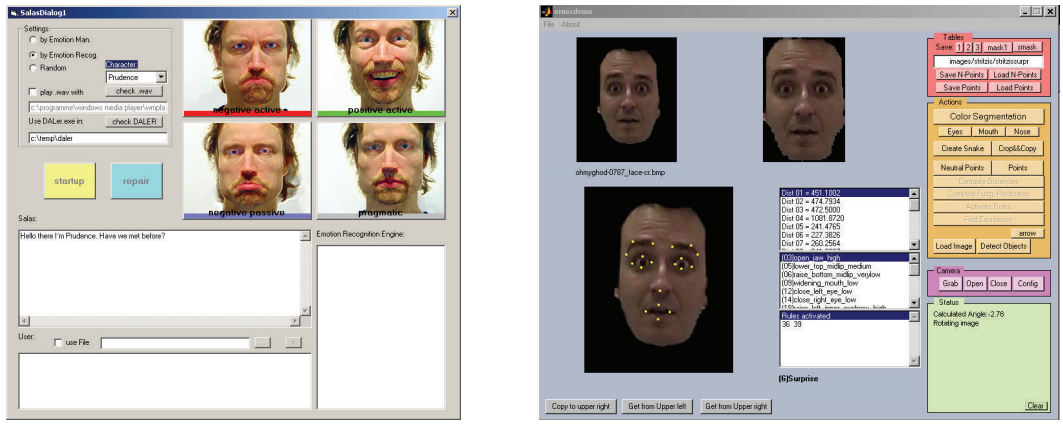
$$t(x_1, x_2, \ldots, x_n) = \min(x_1, x_2, \ldots, x_n), \tag{31}$$

the algebraic product

$$t(x_1, x_2, \ldots, x_n) = x_1 \cdot x_2 \cdot \ldots \cdot x_n, \tag{32}$$

Table 14: Rules with FAP range of variation in MPEG-4 units.

Rule	Quadrant
$F_6 \in [160, 240]$, $F_7 \in [160, 240]$, $F_{12} \in [260, 340]$, $F_{13} \in [260, 340]$, $F_{19} \in [-449, -325]$, $F_{20} \in [-426, -302]$, $F_{21} \in [325, 449]$, $F_{22} \in [302, 426]$, $F_{33} \in [70, 130]$, $F_{34} \in [70, 130]$, $F_{41} \in [130, 170]$, $F_{42} \in [130, 170]$, $F_{53} \in [160, 240]$, $F_{54} \in [160, 240]$	(++)
$F_{16} \in [45, 155]$, $F_{18} \in [45, 155]$, $F_{19} \in [-330, -200]$, $F_{20} \in [-330, -200]$, $F_{31} \in [-200, -80]$, $F_{32} \in [-194, -74]$, $F_{33} \in [-190, -70]$, $F_{34} \in [-190, -70]$, $F_{37} \in [65, 135]$, $F_{38} \in [65, 135]$	(−+)
$F_3 \in [400, 560]$, $F_5 \in [-240, -160]$, $F_{19} \in [-630, -570]$, $F_{20} \in [-630, -570]$, $F_{21} \in [-630, -570]$, $F_{22} \in [-630, -570]$, $F_{31} \in [460, 540]$, $F_{32} \in [460, 540]$, $F_{33} \in [360, 440]$, $F_{34} \in [360, 440]$, $F_{35} \in [260, 340]$, $F_{36} \in [260, 340]$, $F_{37} \in [60, 140]$, $F_{38} \in [60, 140]$	(−+)



(a) (b)

Figure 20: (a) SALAS interaction interface. (b) Facial expression analysis interface.

the bounded sum

$$t(x_1, x_2, \ldots, x_n) = x_1 + x_2 + \cdots + x_n + 1 - n, \tag{33}$$

and so on. Another well-known approach in rule evaluation is described in [55] and utilizes a weighted sum instead of a t-norm in order to combine information from different rule antecedents:

$$y = w_1 x_1 + w_2 x_2 + \cdots + w_n x_n. \tag{34}$$

Both approaches are well studied and established in the field of fuzzy automatic control. Still, they are not adequate for the case of facial expression estimation: their main disadvantage is that they assume that all antecedents are known, that is, that all features are measured successfully and precisely. In the case of facial expression estimation, FAPs may well be estimated with a very low confidence, or not estimated at all, due to low video quality, occlusion, noise, and so on. Thus, a more flexible rule evaluation scheme is required, that is able to incorporate such uncertainty as well. Moreover, the second one of the conventional approaches, due to the summation form, has the disadvantage of possibly providing a highly activated output even in the case that an important antecedent is known to be missing; obviously, it is not suitable for the case examined in this paper, where the non-activation of an FAP automatically implies that the expression profiles that require it are not activated either. For this reason, in this work we have used a flexible rule evaluation scheme [56], which is in fact a generalization of the t-norm-based conventional approach. In this approach and in the t-norm operation described in (30), antecedents with lower values affect most the resulting value of y, while antecedents with values close to one have trivial and negligible affect on the value of y. Having that in mind, we can demand that only antecedents that are known with a high confidence will be allowed to have low values in that operation. Then, the activation level of a rule with this approach can be interpreted in a possibilistic manner, that is, it can be interpreted as the degree to which the corresponding output is possible, according to the available information; in the literature, this possibilistic degree is referred to as plausibility. The confidence is determined by the confidence values of the utilized inputs, that is, by the confidence values of the rule antecedents, as follows:

$$y^c = \frac{x_1^c + x_2^c + \cdots + x_n^c}{n}. \tag{35}$$

7. EXPERIMENTAL RESULTS

7.1. Test data generation: the SALAS-emotion induction framework

Our test data have been produced using the SALAS testbed application developed within the ERMIS and HUMAINE projects, which is an extension of one of the highlights of AI research in the 1960s, Weizenbaum's ELIZA [57]. The ELIZA framework simulates a Rogerian therapy, during which clients talk about their problems to a listener that provides responses that induces further interaction without passing any comment or judgment.

Recording is an integral part of this challenge. With the requirement of both audio and visual inputs, the need to compromise between demands of psychology and signal processing is imminent. If one is too cautious about the recording quality, subjects may feel restrained and are unlikely to show the everyday, relaxed emotionality that would cover most of the emotion representation space. On the other hand, visual and audio analysis algorithms cannot be expected to cope with totally unconstrained head and hand movement, subdued lighting, and mood music. Major issues may also arise from the different requirements of the individual modalities: while head mounted microphones might suit analysis of speech, they can have devastating consequences for visual analysis. Eventually arrangements were developed to ensure that on the visual side, the face was usually almost frontal and well and evenly lit to the human eye; that it was always easy for a human listener to make out what was being said; and that the setting allowed most human participants to relax and express emotion within a reasonable time.

The implementation of SALAS is mainly a software application designed to let a user work through various emotional states. It contains four "personalities" shown in Figure 20(a) that listen to the user and respond to what he/she says, based on the different emotional characteristics that each of the "personalities" possesses. The user controls the emotional tone of the interaction by choosing which "personality" they will interact with, while still being able to change the tone at any time by choosing a different personality to talk to.

The initial recording took place with 20 subjects generating approximately 200 minutes of data. The second set of recordings comprised 4 subjects recording two sessions each, generating 160 minutes of data, providing a total of 360 minutes of data from English speakers; both sets are balanced for gender, 50/50 male/female. These sets provided the input to facial feature extraction and expression recognition system of this paper.

7.2. Facial feature extraction results

Facial feature extraction can be seen as a subcategory of image segmentation, that is, image segmentation into facial features. According to Zhang [58] segmentation algorithms can be evaluated analytically or empirically. Analytical methods directly treat the algorithms themselves by considering the principles, requirements, utilities, complexity, and so forth of algorithms; while these methods can provide an algorithm evaluation which is independent from the implementation itself or the arrangement and choice of input data, very few properties of the algorithm can be obtained or is practical to obtain through analytical study. On the other hand, empirical methods can be divided in two categories: empirical goodness methods, which use a specific "goodness"; measure to evaluate the performance of algorithms, and empirical discrepancy methods which measure the discrepancy between the automatic algorithm result and an ideally labeled image. Zhang reviewed a number of simple discrepancy measures of which, if we consider image segmentation as a pixel classification process, only one is applicable here: the number of misclassified pixels on each facial feature.

While manual feature extraction does not necessarily require expert annotation, it is clear that especially in low-resolution images manual labeling introduces an error. It is therefore desirable to obtain a number of manual interpretations in order to evaluate the interobserver variability. A way to compensate for the latter is Williams' Index (WI) [59], which compares the agreement of an observer with the joint agreement of other observers. An extended version of WI which deals with multivariate data can be found in [60]. The modified Williams' Index I' divides the average number of agreements (inverse disagreements, $D_{j,j'}$) between the computer (observer 0) and $n-1$ human observers (j) by the average number of agreements between human observers:

$$\mathrm{WI} = \frac{(1/n)\sum_{j=1}^{n}(1/D_{0,j})}{(2/n(n-1))\sum_{j}\sum_{j':j'>j}(1/D_{j,j'})}, \tag{36}$$

and in our case we define the average disagreement between two observers j, j' as

$$D_{j,j'} = \frac{1}{D_{\mathrm{bp}}}\left\|M_j^x \veebar M_{j'}^x\right\|, \tag{37}$$

where $\veebar$ denotes the pixel-wise x or operator, $\|M_j^x\|$ denotes the cardinality of feature mask x constructed by observer j, and D_{bp} is used as a normalization factor to compensate for camera zoom on video sequences.

From a dataset of about 50 000 frames, 250 frames were selected at random and the 19 FPs were manually selected from two observers on each one. WI was calculated using (36) for each feature and for each frame separately. At a value of 0, the computer mask is infinitely far from the observer mask. When WI is larger than 1, the computer generated mask disagrees less with the observers than the observers disagree with each other. Distribution of the average WI calculated over the two eyes and mouth for each frame is shown in Figure 21, while Figure 22 depicts the average WI calculated on the two eyebrows. Table 15 summarizes the results.

For the eyes and mouth, WI has been calculated for both the final mask and each of the intermediate masks. WI_x denotes WI for single mask x and WI_f is the WI for the final mask for each facial feature; $\langle \mathrm{WI}_x \rangle$ denotes the average WI for mask x calculated over all test frames.

Column 7 of Table 15 shows the percentage of frames where the mask fusion resulted in an improvement of the WI, while columns 8 and 9 display the average WI in the frames

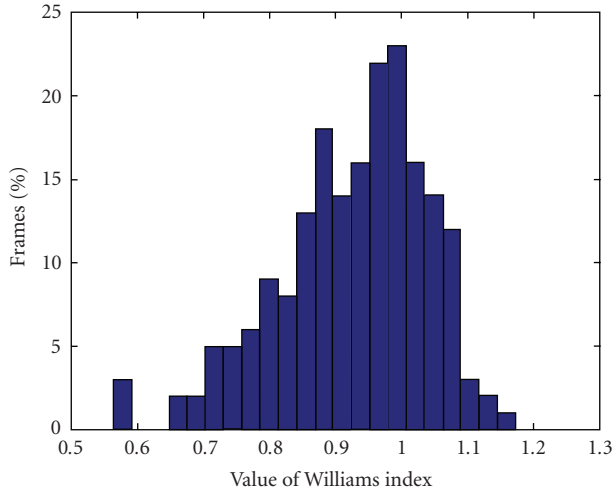


Figure 21: Williams index distribution (average on eyes and mouth).

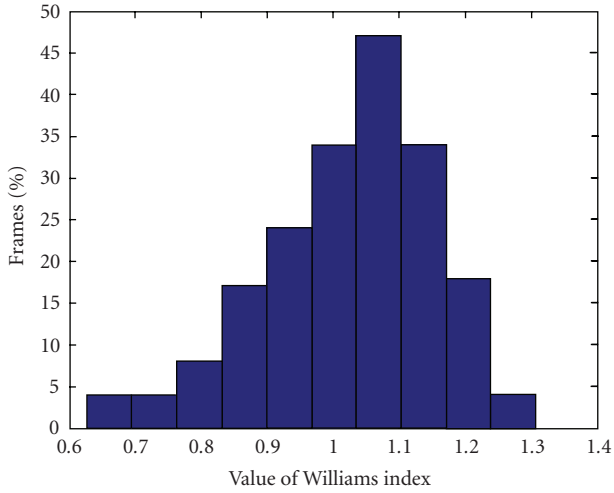


Figure 22: Williams index distribution (average on left and right eyebrows).

where the fusion result was better and worse from the single mask, respectively. One may be tempted to deduct from this table that some feature detectors perform better than the combined mask result; it may seem so when considering the average values, but this is not the case when examining each frame: different methods perform better for different input images and the average results that seem to favor some methods over the others are dependent on the selection of the input frames. This may be also justified especially by looking at the result variation between the left and right eyes for the same mask, as well as from the values of column 8: the average *WI* on frames where eye mask 4 performed better than the fused result is still a bit lower than the total average *WI*, thus it may seem that this mask performs better for the specific test but the improvement is not significant; this means that even when considering the same sequence, the average values may be slightly better for one mask but relying solely on this one mask, the system will have no safeguard to refer to when the algorithm resulting in this mask performs poorly. The latter is demonstrated in column 10 where one can see that when using the fused masks, the worst cases will be on average better than the worse case of the mask with the best mean *WI*. Nevertheless, the aim of this work is not to find the best feature extractor, but to combine them intelligently with respect to the input video. What can be deducted about the different masks is that looking at the value differences between column 8 and column 9, one can conclude that for example eye mask 0 performs better in "very difficult" test frames, where the total average *WI* has a value of 0.69.

7.3. Expression analysis results

Since the ERMIS dataset was created by engaging participants to emotional dialogue, facial expressions in these video sequences were not acted and extreme, but are mostly naturalistic. We evaluated sequences totalling about 30 000 frames. Expression analysis results were tested against manual multimodal annotation from experts [61] and the results are presented in Table 16.

In order to produce the facial expression analysis results, we utilized the neurofuzzy network presented in [53]. The architecture of this network was able to exploit not only FAPs values produced by tracking the feature points and their distances, but also the confidence measures associated with each intermediate result. Since we are dealing with video sequences depicting human-computer interaction, expressivity, head movement and rotation are usually unconstrained. As a result, exact feature point localization is not always possible due to changing lighting conditions, such as varying shadow artifacts introduced by the eyebrow protrusion or the nose. It is given that the contribution of the algorithm presented here lies not only in the fact that it performs stable feature point localization, but more importantly in the fusion process and the confidence measure that it produces for each mask, as well as the fused result. The confidence measure is utilized by the neurofuzzy network to reduce the importance of a set of FAP measurements in a frame where confidence is low, thereby catering for better network training and adaptation, since the network is trained with examples that perform better. The significance of this approach is proven with the increase in performance shown in [53], as well as in the second column of Table 16, where the possibilistic approach [56], which also utilizes the confidence measure, also outperforms a "naive" fuzzy rule implementation based only on FAP values.

In addition to this, Column 5 in Table 15 indicates that the fusion step almost always improves the performance of the individual masks, in the sense that it produces a final result which agrees more with the expert annotators than in the case of the single masks (higher Williams Index value, which produces a ratio of the fused mask over the single masks > 1). The robustness of the feature extraction process, when combined with the provision of confidence measures, is shown in the videos at http://www.image.ece.ntua.gr/ijivp. These videos contain the results from the feature extraction process per frame, and the estimated quadrant which con-

Table 15: Result summary.

Algorithm[1]	Mask #	$\langle \mathrm{WI}_x \rangle$	$\langle \mathrm{WI}_f \rangle$	$\frac{\langle \mathrm{WI} \rangle_f}{\langle \mathrm{WI} \rangle_x}$	σ^2	% of frames where $\mathrm{WI}_f > \mathrm{WI}_x$	$\langle \mathrm{WI} \rangle$ in frames where $\mathrm{WI}_f < \mathrm{WI}_x$	$\langle \mathrm{WI} \rangle$ in frames where $\mathrm{WI}_x < \mathrm{WI}_f$	$\langle \mathrm{WI} \rangle$ in 5% worst frames[4]
1	2	3	4	5	6	7	8	9	10
Left Eye									
NN[2]	0	0.677	—	1.287	0.103	74.2	0.697	0.885	0.351
Section 4.1.1	1	0.701	—	1.216	0.056	78.8	0.731	0.868	0.414
Section 4.1.2	2	0.821	0.838	1.029	0.027	82.4	0.770	0.887	0.459
Section 4.1.3	3	0.741	—	1.131	0.057	76.2	0.811	0.847	0.265
Section 4.1.4	4	0.870	—	0.979	0.026	44.3	0.812	0.867	0.427
	f	0.838	—	1.000	—	—	—	—	0.475
Right Eye									
NN[2]	0	0.800	—	1.093	0.020	75.2	0.672	0.946	0.411
Section 4.1.1	1	0.718	—	1.243	0.084	81.4	0.674	0.929	0.352
Section 4.1.2	2	0.774	0.875	1.140	0.021	58.2	0.836	0.883	0.396
Section 4.1.3	3	0.650	—	1.346	0.028	84.5	0.632	0.920	0.305
Section 4.1.4	4	0.893	—	0.982	0.02	48.4	0.778	0.996	0.418
	f	0.875	—	1.000	—	—	—	—	0.429
Mouth									
Section 4.4.1	1	0.763	—	1.051	0.046	59.2	0.752	0.772	0.288
Section 4.4.2	2	0.823	0.780	0.963	0.038	44.8	0.721	0.852	0.345
Section 4.4.3	3	0.570	—	1.446	0.204	96.9	0.510	0.793	0.220
	f	0.780	—	1.000	—	—	—	—	0.359
Eyebrows[3]									
Left		1.034	—	—	—	—	—	—	—
Right		1.013	—	—	—	—	—	—	—

WI_x denotes WI for single mask x and WI_f is the WI for the final mask for each facial feature.
$\langle \bullet \rangle$ denotes the average over all features in all frames, $\langle \bullet \rangle_f$ denotes the average of the final masks over all frames while $\langle \bullet \rangle_x$ denotes the average of mask x over all frames.
[1] Refer to indicated subsection number
[2] NN denotes $\mathbf{M}^e_{\mathrm{nn}}$, the eye mask derived directly from the neural network output
[3] Using eyebrow mask $\mathbf{M}^b_{E_2}$, prior to thinning
[4] $\langle \mathrm{WI} \rangle$ in the 5% of total frames with the lowest *WI*.

Table 16: Comparison of results between manual and two automatic expression analysis approaches.

Naive fuzzy rules	Possibilistic approach	Annotator disagreement
65.1%	78.4%	20.01%

tains the observed facial expression. Even though feature localization may be inaccurate or even fail in specific frames, this fact is identified by a low-confidence measure, effectively instructing the expression analysis algorithm to ignore these features and try to estimate the facial expression on the remaining results.

As a general rule, the last column of Table 16 indicates that the human experts that classify the frames to generate the ground truth make contrasting evaluations once every five frames; this fact is clearly indicative of the ambiguity of the observed emotions in a naturalistic environment. It is also worth underlining that this system achieves a 78% classification rate while operating based solely on expert knowledge provided by humans in the form of fuzzy rules, without weights for the rule antecedents. Allowing for the specification of antecedence importance as well as for rule optimization through machine learning is expected to provide for even further enhancement of the achieved results.

8. CONCLUSIONS

In this work we have presented a method to automatically locate 19 facial feature points that are used in combination with the MPEG-4 facial model for expression estimation. A robust method for locating these features has been presented which also extracts a confidence estimate depicting a "goodness" measure of each detected point, which is used by the expression recognition stage; the provision of this measure enables the expression recognition process to discard falsely located features, thus enhancing performance in recognizing both universal (basic) emotion labels, as well as intermediate expressions based on a dimensional representation. Our algorithm can perform well under a large variation of facial image quality, color, and resolution.

Since the proposed method only handles roll facial rotation, an extension to be considered is the incorporation of a facial model. Recently, a lot of work has been done in facial feature detection and fitting of facial models [62]. While these techniques can detect facial features, but not extract their precise boundary, they can extend our work by accurately predicting the face position in each frame. Thus, feature candidate areas would be defined with greater precision allowing the system to work even under large head rotation and feature occlusion.

REFERENCES

[1] A. Mehrabian, "Communication without words," *Psychology Today*, vol. 2, no. 9, pp. 52–55, 1968.

[2] B. Fasel and J. Luettin, "Automatic facial expression analysis: a survey," *Pattern Recognition*, vol. 36, no. 1, pp. 259–275, 2003.

[3] P. Ekman and W. V. Friesen, *Facial Action Coding Systems: A Technique for the Measurement of Facial Movement*, Consulting Psychologist Press, Palo Alto, Calif, USA, 1978.

[4] C. Tomasi and T. Kanade, "Detection and tracking of point features," Tech. Rep. CMU-CS-91-132, Carnegie Mellon University, Pittsburgh, Pa, USA, April 1991.

[5] Y.-L. Tian, T. Kanade, and J. F. Cohn, "Recognizing action units for facial expression analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 2, pp. 97–115, 2001.

[6] A. M. Tekalp and J. Ostermann, "Face and 2-D mesh animation in MPEG-4," *Signal Processing: Image Communication*, vol. 15, no. 4, pp. 387–421, 2000.

[7] A. Raouzaiou, N. Tsapatsoulis, K. Karpouzis, and S. Kollias, "Parameterized facial expression synthesis based on MPEG-4," *EURASIP Journal on Applied Signal Processing*, vol. 2002, no. 10, pp. 1021–1038, 2002.

[8] I. A. Essa and A. P. Pentland, "Coding, analysis, interpretation, and recognition of facial expressions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 757–763, 1997.

[9] A. Lanitis, C. J. Taylor, T. F. Cootes, and T. Ahmed, "Automatic interpretation of human faces and hand gestures using flexible models," in *Proceedings of the 1st International Workshop on Automatic Face and Gesture Recognition (FG '95)*, pp. 98–103, Zurich, Switzerland, September 1995.

[10] Y. Yacoob and L. S. Devis, "Recognizing human facial expressions from long image sequences using optical flow," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 6, pp. 636–642, 1996.

[11] C. L. Lisetti and D. E. Rumelhart, "Facial expression recognition using a neural network," in *Proceedings of the 11th International Florida Artificial Intelligence Research Society Conference*, pp. 328–332, AAAI Press, Sanibel Island, Fla, USA, May 1998.

[12] S. Kaiser and T. Wehrle, "Automated coding of facial behavior in human-computer interactions with facs," *Journal of Nonverbal Behavior*, vol. 16, no. 2, pp. 67–84, 1992.

[13] G J. Edwards, T. F. Cootes, and C. J. Taylor, "Face recognition using active appearance models," in *Proceedings of the 5th European Conference on Computer Vision (ECCV '98)*, vol. 2, pp. 581–595, Freiburg, Germany, June 1998.

[14] J. F. Cohn, A. J. Zlochower, J. J. Lien, and T. Kanade, "Feature-point tracking by optical flow discriminates subtle differences in facial expression," in *Proceedings of the 3rd IEEE International Conference on Automatic Face and Gesture Recognition (FG '98)*, pp. 396–401, Nara, Japan, April 1998.

[15] M. J. Black and Y. Yacoob, "Recognizing facial expressions in image sequences using local parameterized models of image motion," *International Journal of Computer Vision*, vol. 25, no. 1, pp. 23–48, 1997.

[16] K.-M. Lam and H. Yan, "An analytic-to-holistic approach for face recognition based on a single frontal view," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 7, pp. 673–686, 1998.

[17] H. Gu, G.-D. Su, and C. Du, "Feature points extraction from face images," in *Proceedings of the Image and Vision Computing Conference (IVCNZ '03)*, pp. 154–158, Palmerston North, New Zealand, November 2003.

[18] S.-H. Leung, S.-L. Wang, and W.-H. Lau, "Lip image segmentation using fuzzy clustering incorporating an elliptic shape function," *IEEE Transactions on Image Processing*, vol. 13, no. 1, pp. 51–62, 2004.

[19] N. Sarris, N. Grammalidis, and M. G. Strintzis, "FAP extraction using three-dimensional motion estimation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 10, pp. 865–876, 2002.

[20] Y. Tian, T. Kanade, and J. F. Cohn, "Robust lip tracking by combining shape, color and motion," in *Proceedings of the 4th Asian Conference on Computer Vision (ACCV '00)*, pp. 1040–1045, Taipei, Taiwan, January 2000.

[21] N. Sebe, M. S. Lew, I. Cohen, Y. Sun, T. Gevers, and T. S. Huang, "Authentic facial expression analysis," in *Proceedings of the 6th IEEE International Conference on Automatic Face and Gesture Recognition (FG '04)*, pp. 517–522, Seoul, South Korea, May 2004.

[22] D. DeCarlo and D. Metaxas, "The integration of optical flow and deformable models with applications to human face shape and motion estimation," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '96)*, pp. 231–238, San Francisco, Calif, USA, June 1996.

[23] M. Pantic and L. J. M. Rothkrantz, "Expert system for automatic analysis of facial expressions," *Image and Vision Computing*, vol. 18, no. 11, pp. 881–905, 2000.

[24] T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active appearance models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 6, pp. 681–685, 2001.

[25] C.-L. Huang and Y.-M. Huang, "Facial expression recognition using model-based feature extraction and action parameters classification," *Journal of Visual Communication and Image Representation*, vol. 8, no. 3, pp. 278–290, 1997.

[26] M. J. Lyons, J. Budynek, and S. Akamatsu, "Automatic classification of single facial images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 12, pp. 1357–1362, 1999.

[27] H. Hong, H. Neven, and C. von der Malsburg, "Online facial expression recognition based on personalized galleries," in *Proceedings of the 3rd IEEE International Conference on Automatic Face and Gesture Recognition (FG '98)*, pp. 354–359, Nara, Japan, April 1998.

[28] R.-L. Hsu, M. Abdel-Mottaleb, and A. K. Jain, "Face detection in color images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 696–706, 2002.

[29] S. J. McKenna, Y. Raja, and S. Gong, "Tracking colour objects using adaptive mixture models," *Image and Vision Computing*, vol. 17, no. 3-4, pp. 225–231, 1999.

[30] ERMIS, "Emotionally Rich Man-machine Intelligent System IST-2000-29319," http://www.image.ntua.gr/ermis/.

[31] HUMAINE IST, "Human-Machine Interaction Network on Emotion," 2004–2007, http://www.emotion-research.net/.

[32] ISTFACE, "MPEG-4 Facial Animation System—Version 3.3.1 Gabriel Abrantes," (Developed in the context of the European Project ACTS MoMuSys 97-98 Instituto Superior Tecnico).

[33] J. W. Young, "Head and Face Anthropometry of Adult U.S. Civilians," FAA Civil Aeromedical Institute, 1963–1993, (final report 1993).

[34] M.-H. Yang, D. J. Kriegman, and N. Ahuja, "Detecting faces in images: a survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 1, pp. 34–58, 2002.

[35] C. P. Papageorgiou, M. Oren, and T. Poggio, "A general framework for object detection," in *Proceedings of the 6th IEEE International Conference on Computer Vision (ICCV '98)*, pp. 555–562, Bombay, India, January 1998.

[36] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '01)*, vol. 1, pp. 511–518, Kauai, Hawaii, USA, December 2001.

[37] I. Fasel, B. Fortenberry, and J. Movellan, "A generative framework for real time object detection and classification," *Computer Vision and Image Understanding*, vol. 98, no. 1, pp. 182–210, 2005.

[38] R. Fransens, J. De Prins, and L. van Gool, "SVM-based nonparametric discriminant analysis, an application to face detection," in *Proceedings of the 9th IEEE International Conference on Computer Vision (ICCV '03)*, vol. 2, pp. 1289–1296, Nice, France, October 2003.

[39] S. Kollias and D. Anastassiou, "An adaptive least squares algorithm for the efficient training of artificial neural networks," *IEEE Transactions on Circuits and Systems*, vol. 36, no. 8, pp. 1092–1101, 1989.

[40] M. H. Hagan and M. B. Menhaj, "Training feedforward networks with the Marquardt algorithm," *IEEE Transactions on Neural Networks*, vol. 5, no. 6, pp. 989–993, 1994.

[41] L. Yin and A. Basu, "Generating realistic facial expressions with wrinkles for model-based coding," *Computer Vision and Image Understanding*, vol. 84, no. 2, pp. 201–240, 2001.

[42] M. J. Lyons, M. Haehnel, and N. Tetsutani, "The mouthesizer: a facial gesture musical interface," in *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '01)*, p. 230, Los Angeles, Calif, USA, August 2001.

[43] S. Arca, P. Campadelli, and R. Lanzarotti, "An automatic feature-based face recognition system," in *Proceedings of the 5th International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS '04)*, Lisboa, Portugal, April 2004.

[44] K.-M. Lam and H. Yan, "Locating and extracting the eye in human face images," *Pattern Recognition*, vol. 29, no. 5, pp. 771–779, 1996.

[45] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679–698, 1986.

[46] D. O. Gorodnichy, "On importance of nose for face tracking," in *Proceedings of the 5th IEEE International Conference on Automatic Face and Gesture Recognition (FG '02)*, pp. 181–186, Washington, DC, USA, May 2002.

[47] S. C. Aung, R. C. K. Ngim, and S. T. Lee, "Evaluation of the laser scanner as a surface measuring tool and its accuracy compared with direct facial anthropometric measurements," *British Journal of Plastic Surgery*, vol. 48, no. 8, pp. 551–558, 1995.

[48] L. Vincent, "Morphological grayscale reconstruction in image analysis: applications and efficient algorithms," *IEEE Transactions on Image Processing*, vol. 2, no. 2, pp. 176–201, 1993.

[49] L. Vincent, "Morphological grayscale reconstruction in image analysis: applications and efficient algorithms," *IEEE Transactions on Image Processing*, vol. 2, no. 2, pp. 176–201, 1993.

[50] A. Krogh and J. Vedelsby, "Neural network ensembles, cross validation, and active learning," in *Advances in Neural Information Processing Systems*, G. Tesauro, D. Touretzky, and T. Leen, Eds., vol. 7, pp. 231–238, The MIT Press, Cambridge, Mass, USA, 1995.

[51] V. Tresp, "Committee machines," in *Handbook for Neural Network Signal Processing*, Y. H. Hu and J.-N. Hwang, Eds., CRC Press, Boca Raton, Fla, USA, 2001.

[52] C. M. Whissel, "The dictionary of affect in language," in *Emotion: Theory, Research and Experience. The Measurement of Emotions*, R. Plutchnik and H. Kellerman, Eds., vol. 4, pp. 113–131, Academic Press, New York, NY, USA, 1989.

[53] S. Ioannou, A. T. Raouzaiou, V. A. Tzouvaras, T. P. Mailis, K. Karpouzis, and S. Kollias, "Emotion recognition through facial expression analysis based on a neurofuzzy network," *Neural Networks*, vol. 18, no. 4, pp. 423–435, 2005.

[54] G. J. Klir and B. Yuan, *Fuzzy Sets and Fuzzy Logic: Theory and Applications*, Prentice-Hall, Upper Saddle River, NJ, USA, 1995.

[55] M. A. Lee and H. Takagi, "Integrating design stages of fuzzy systems using genetic algorithms," in *Proceedings of the 2nd IEEE International Conference on Fuzzy Systems (FUZZY'93)*, pp. 612–617, San Francisco, Calif, USA, March-April 1993.

[56] M. Wallace and S. Kollias, "Possibilistic evaluation of extended fuzzy rules in the presence of uncertainty," in *Proceedings of the 14th IEEE International Conference on Fuzzy Systems (FUZZ '05)*, pp. 815–820, Reno, Nev, USA, May 2005.

[57] J. Weizenbaum, "ELIZA—a computer program for the study of natural language communication between man and machine," *Communications of the ACM*, vol. 9, no. 1, pp. 36–45, 1966.

[58] Y. J. Zhang, "A survey on evaluation methods for image segmentation," *Pattern Recognition*, vol. 29, no. 8, pp. 1335–1346, 1996.

[59] G. W. Williams, "Comparing the joint agreement of several raters with another rater," *Biometrics*, vol. 32, no. 3, pp. 619–627, 1976.

[60] V. Chalana and Y. Kim, "A methodology for evaluation of boundary detection algorithms on medical images," *IEEE Transactions on Medical Imaging*, vol. 16, no. 5, pp. 642–652, 1997.

[61] R. Cowie, E. Douglas-Cowie, S. Savvidou, E. McMahon, M. Sawey, and M. Schröder, "'Feeltrace': an instrument for recording perceived emotion in real time," in *Proceedings of the ISCA Workshop on Speech and Emotion*, pp. 19–24, Belfast, Northern Ireland, September 2000.

[62] D. Cristinacce and T. F. Cootes, "A comparison of shape constrained facial feature detectors," in *Proceedings of the 6th IEEE International Conference on Automatic Face and Gesture Recognition (FG '04)*, pp. 375–380, Seoul, South Korea, May 2004.

Hindawi Publishing Corporation
EURASIP Journal on Image and Video Processing
Volume 2007, Article ID 81387, 8 pages
doi:10.1155/2007/81387

Research Article

Real-Time 3D Face Acquisition Using Reconfigurable Hybrid Architecture

Johel Mitéran, Jean-Philippe Zimmer, Michel Paindavoine, and Julien Dubois

Le2i Laboratory, University of Burgundy, BP 47870, 21078 DIJON Cedex, France

Received 2 May 2006; Revised 22 November 2006; Accepted 12 December 2006

Recommended by Joern Ostermann

Acquiring 3D data of human face is a general problem which can be applied in face recognition, virtual reality, and many other applications. It can be solved using stereovision. This technique consists in acquiring data in three dimensions from two cameras. The aim is to implement an algorithmic chain which makes it possible to obtain a three-dimensional space from two two-dimensional spaces: two images coming from the two cameras. Several implementations have already been considered. We propose a new simple real-time implementation based on a hybrid architecture (FPGA-DSP), allowing to consider an embedded and reconfigurable processing. Then we show our method which provides depth map of face, dense and reliable, and which can be implemented on an embedded architecture. A various architecture study led us to a judicious choice allowing to obtain the desired result. The real-time data processing is implemented in an embedded architecture. We obtain a dense face disparity map, precise enough for considered applications (multimedia, virtual worlds, biometrics) and using a reliable method.

1. INTRODUCTION

We present in this paper a comparison of numerous methods allowing obtaining a dense depth map of human face, and the real-time implementation of the chosen method. Acquiring 3D data of human face is a general problem which can be applied in face recognition [1–3]. In this particular case, the knowledge of depth map can be used for example as a classification feature. It can be seen as an improvement of classical method such as eigenfaces [4]. The stereovision technique we used is well known and consists in acquiring data in three dimensions from two cameras. The key problem in stereo is how to find the corresponding points in the left and in the right image [5] (correspondence problem). Many research activities are currently dealing with stereovision, using different approaches to solve the correspondence problem. Since our main application is face recognition, we studied different methods adapted to this problem. Moreover, our applications have to be completed in real-time (10 image/s). General purpose computers are not fast enough to meet these requirements because of the algorithmic complexity of stereovision techniques. We studied the implementation using hybrid approach. Although various implementations have already been considered [6, 7], we propose a simple real-time implementation, including a regularization step, based on a multiprocessor approach (FPGA-DSP) allowing to consider an embedded and reconfigurable processing. Faugeras et al. [6] proposed a multi-FPGA (23 Xilinx 3090) architecture which is too complex for an embedded application. Ohm and Izquierdo [7] proposed a stereo algorithm where dense map is obtained using bilinear interpolation from global disparity estimation. However, this approach used for face localization is not enough precise for face recognition problem. In [8], Porr et al. used the Gabor-based method implemented in a software and hardware system. The board is virtex-based as ours, but does not allow embedded post processing as we do in the DSP from Texas Instrument. We present in the first part of the paper the study of the whole necessary processing, while reviewing and comparing various employed methods. In the second part, we present the implementation on an embedded architecture of our method which provides depth map of face, dense and reliable.

2. METHOD

2.1. Stereodata processing flow

The main goal of this whole processing is to match corresponding points between two images. The distance or disparity between these homologous points is then calculated.

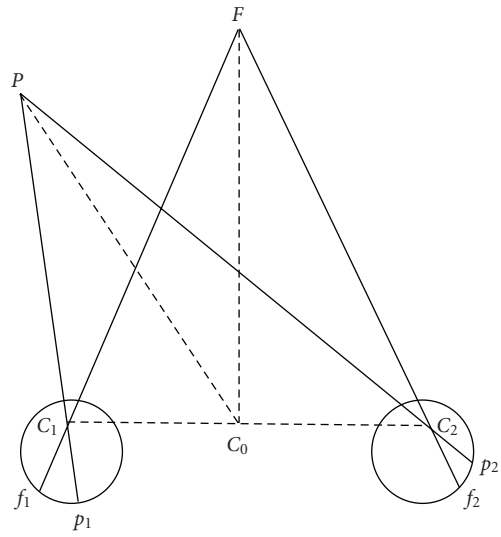


Figure 1: Retina disparity.

This value is proportional to the depth, thus codes the third dimension (Figure 1).

The retina disparity D is defined as follows:

$$D = E(f_1, p_1) - E(f_2, p_2), \tag{1}$$

where $E(x, y)$ is the Euclidian distance between x and y.

This value is proportional to the depth difference between P and F.

The processing flow is composed of two main parts. The first requires mainly geometrical criteria and modeling, the second uses signal processing knowledge.

The first part is the camera calibration, either for each one in 3D space (strong calibration) or relatively between them (weak calibration and epipolar geometry). To this stage can be added a rectification image processing. This rectification allows to match the image lines of the stereo pair, and thus to work in only one dimension [5].

The second part consists in the homologous points matching. Various methods were developed to constitute dense depth maps. Two papers [9, 10] present a large review of these techniques. Since the goal of this paper is mainly to present the hardware implementation of our solution, we will only recall the principle of the 3 methods we compared, the results of this comparison which will justify our final choice.

2.2. Principal methods of dense depth maps constitution

Several methods have been studied and give interesting results. We can classify them in three principal parts: the methods based on partial differential equation (PDE) [11], on local phase [12], and on crosscorrelation [13].

2.2.1. Partial differential equations

This method is based on the minimization of an energy criterion by solving a diffusion equation. Various implementations were given. One of them provides the depth by resolution of the discrete Euler-Lagrange equation [14]. A judicious choice of the regularization function allows preserving discontinuities [11]. A multiresolution result is obtained by iteratively searching for the solution. In order to obtain efficient solutions, it can be here interesting to introduce the epipolar constraint.

The methods based on PDE allow obtaining dense depth maps and a very good precision on the results. Unfortunately, these processing require too significant computing times and cannot, yet, be considered on a simple embedded architecture. Therefore, we did not include this method in our comparison.

2.2.2. Crosscorrelation

This classical method is based on homologous points matching by search of the minimum of a criterion by crosscorrelation in shifting local windows [13]. The most usual criteria used are the crosscorrelation or the square difference (or the difference absolute value) of the pixel intensities between each image of the stereopair. This method can be improved, in order to make it less sensitive to the differences between the average gray level of the two images, by centering and/or by a local normalization. Moreover, the criterion is applied in a local window surrounding the tested pixels. The criterion $C_{x,y}$ is then computed as follows:

$$C_{x,y} = \sum_{-l\le i\le l;\, -h\le j\le h} \left(\left(I_1(x+i, y+j) - \overline{I}_1(x, y) \right) - \left(I_2(x+i, y+j) - \overline{I}_2(x, y) \right) \right)^2, \tag{2}$$

where $I_1(x, y)$ is the pixel luminance of left image, $I_2(x, y)$ is the pixel luminance of right image, and h and l are, respectively, the height and length of the local window centred in (x, y). $\overline{I}_1(x, y)$ and $\overline{I}_2(x, y)$ are the mean of luminance computed in these local windows.

The method of shifting window processing requires a range of limited disparities $[d_1, d_2]$. The criterion is then calculated for each disparity. The maximum criterion gives the required disparity. If the maximum is obtained for d_1 or d_2, an error value is affected to D (Figure 2).

This processing is carried out effectively in one dimension and thus requires either to know the epipolar constraint, or to work on rectified images. A double processing Left Image/Right Image then Right Image/Left Image, followed by a validation step, makes it possible to remove wrong matching.

A multiscale approach can also be considered, allowing an extension of the range of the required disparities and a validation at various scales in order to obtain better results on poorly textured patterns. Improvements were planned in order to obtain better answers in the presence of local discontinuities. Fusiello et al. [15] uses several local windows around the pixel. Devernay [5] uses a local window in form of parallelogram, and deforms it to obtain a minimum

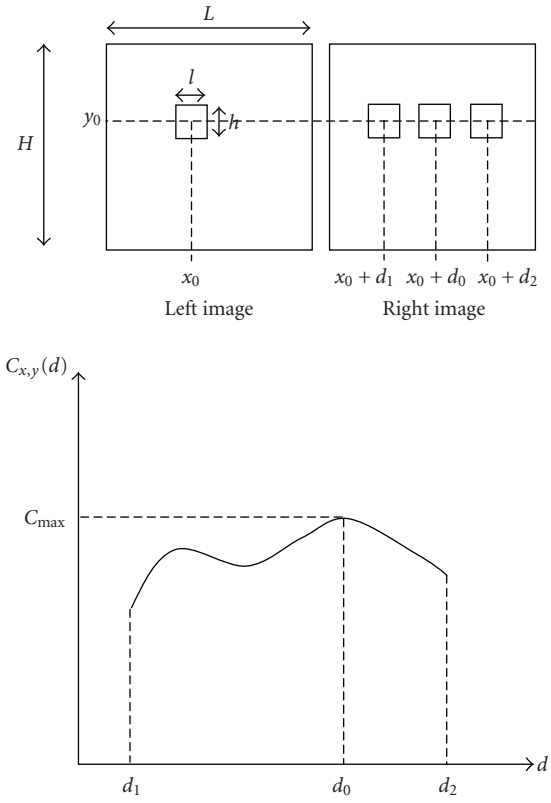


Figure 2: Crosscorrelation-based matching.

criterion. These two methods allow introducing local disparity gradients.

In our case, we improve the correlation results during a regularization step composed by a parabolic approximation of the correlation (allowing subpixel interpolation) and a morphological filtering which allows removing artifacts. The parabolic interpolation is given by

$$d(x,y) = d_0(x,y) + \frac{1}{2}\frac{C_{x,y}(d_0+1) - C_{x,y}(d_0-1)}{2C_{\max} - C_{x,y}(d_0+1) - C_{x,y}(d_0-1)}. \tag{3}$$

2.2.3. Local phase

The algorithm uses the image local phases estimates for the disparity determination [13]. Phase differences, phase derivative, and local frequencies are calculated by filtering the stereocouple with Gabor filters, as follows:

$$\begin{aligned} I_{1G}(x) &= I_1(x) * G(x,\sigma,\omega), \\ I_{2G}(x) &= I_2(x) * G(x,\sigma,\omega), \end{aligned} \tag{4}$$

with the Gabor kernel defined as

$$G(x,\sigma,\omega) = \frac{1}{\sqrt{2\pi\sigma}} e^{-x^2/(2\pi\alpha)^2}, \tag{5}$$

and the local phases are defined as

$$\begin{aligned} \Phi_1(x) &= \arctan\left(\frac{\mathrm{Im}\,(I_{1G}(x))}{\mathrm{Re}\,(I_{1G}(x))}\right), \\ \Phi_2(x) &= \arctan\left(\frac{\mathrm{Im}\,(I_{2G}(x))}{\mathrm{Re}\,(I_{2G}(x))}\right). \end{aligned} \tag{6}$$

The disparity d is calculated from estimates of local phases in images I_1 and I_2 using

$$d_\omega(x) = \frac{(\Phi_1(x) - \Phi_2(x))}{\overline{\omega}}, \tag{7}$$

where $\overline{\omega}$ is the average local spatial frequency.

The processing allows then to deduce local disparities [16]. The frequency scale limitations and the phase wrapping problem impose to limit the disparity. To obtain a higher range of disparity, it is necessary to resort to a coarse-fine strategy in which the results for each scale are extended and used on the following scale, thus making it possible to increase the limits of disparity variations [17]. A regularization step introduces a smoothing constraint for each scale by fitting the results to a spline surface. These methods are related to recent discoveries in physiology of three-dimensional perception [18, 19].

Another method based on local phase determination uses complex wavelets [20]. Through its robustness against lighting variation and additive noise, this method extends the properties of the Gabor wavelets to the differences in luminosity variation and to additive noise. But especially this operator provides shift invariance and a good directional selectivity. These conditions are essential to obtain disparity. The disparity computation is carried out by a difference between the detail coefficients of the left and right images. An adjustment by a least square method gives an optimal disparity, depending on the phase, and insensitive to intensity changes [21]. The epipolar constraint can be added effectively for a better determination of homologous points [22].

2.3. Methods comparison in the case of face acquisition

In order to choose a good compromise between performances and speed processing, we measured the quadratic error between a model of face and the stereo acquisition.

The error is defined as

$$Q = \frac{1}{LH}\sum_{y=1}^{H}\sum_{x=1}^{L} |\,O(x,y) - S(x,y)\,|^2, \tag{8}$$

where $O(x,y)$ represents the depth map obtained using our algorithms and $S(x,y)$ is the model depth map, obtained using a 3D laser-based scanner.

The face used for the comparison is depicted in Figure 3.

We studied the error depending on the focal length used during acquisition. We showed in [23] that the optimum choice for the stereo device depends on the focal length and that this optimum can be chosen around $f = 30$ mm for a standard CCD-based camera.

Rectified images of test face

Reference depth map

Figure 3: Reference face.

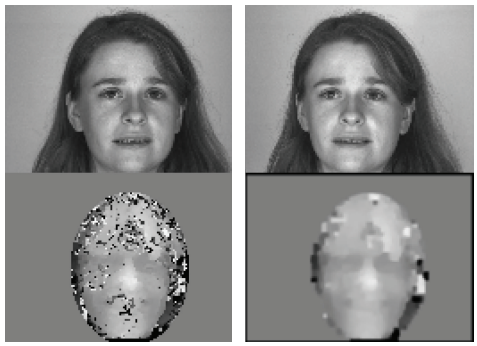
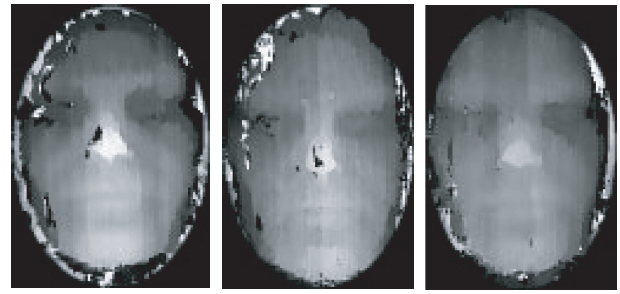


Figure 4: Left and right acquired images, depth maps without and with post processing.

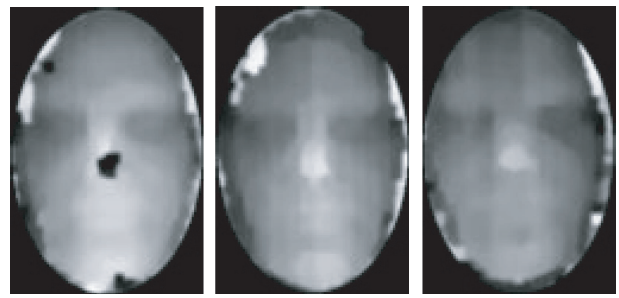
The maps obtained by crosscorrelation can be very correct, under certain conditions of illumination. For our part, we obtained good dense depth map by projecting a random texture on the face. Nevertheless, a post processing is required in order to effectively improve the existing discontinuities. This processing can be filled by a morphological opening and closing, followed by a Gaussian blur to smooth small discontinuities correctly. Figure 4 shows the results obtained with and without filtering.

The images obtained using the three compared methods are depicted on Figure 5, and the corresponding error is depicted on Figure 6. It is clear that, although the Gabor wavelets-based method seems to be the best choice, the performances are very close from each other when focal length is near $f = 30$ mm. This justifies our final choice of implementation, based on the crosscorrelation algorithm, for which the



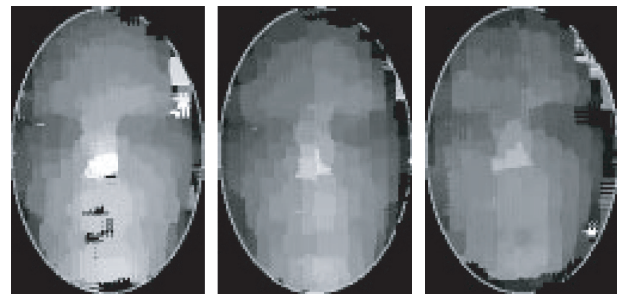
(I) $f = 28.5$ mm (II) $f = 32$ mm (III) $f = 35.3$ mm

(a) Results using crosscorrelation



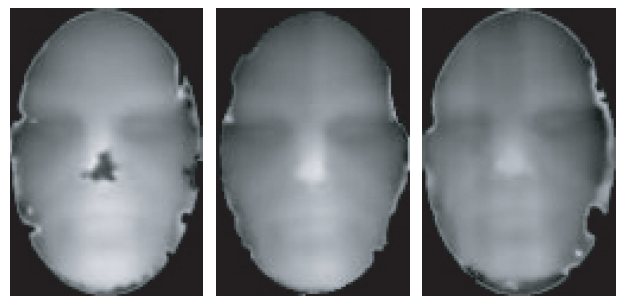
(I) $f = 28.5$ mm (II) $f = 32$ mm (III) $f = 35.3$ mm

(b) Results using filtered crosscorrelation



(I) $f = 28.5$ mm (II) $f = 32$ mm (III) $f = 35.3$ mm

(c) Correlation using multiple windows



(I) $f = 28.5$ mm (II) $f = 32$ mm (III) $f = 35.3$ mm

(d) Results using Gabor wavelets

Figure 5: Depth maps.

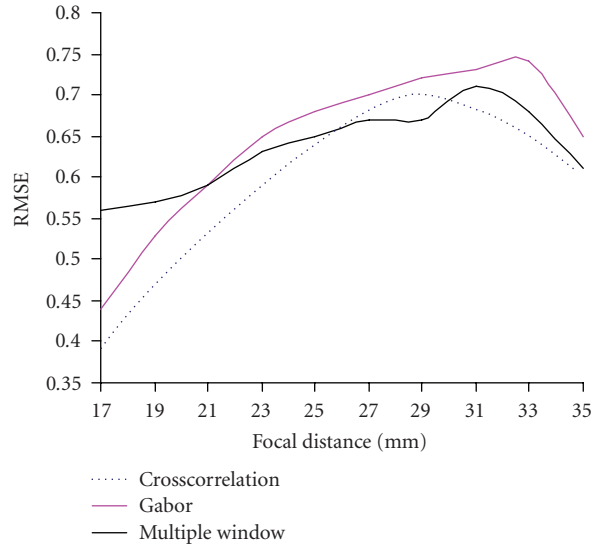


Figure 6: Error comparison.

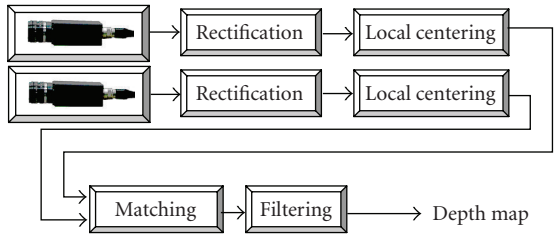


Figure 7: Computed chain implemented to obtain dense depth map.

hardware implement cost will be clearly lower than in the Gabor wavelets case.

3. PROCESSING, RESULTS, AND IMPLEMENTATION

Since we obtained using software simulation dense and precise depth maps, we implemented the crosscorrelation by shifting windows algorithm. The whole algorithm is distributed as shown in Algorithm 1, and is depicted in Figure 7.

In the first stage after image acquisition, we carry out an image rectification. This processing is computed through a weak calibration and the fundamental matrix determination [24]. The rectification matrices are obtained by an original computation, based on a projective method [25], by calculating the homography of four points of each image plan.

We carry out then a local centering of the images thus allowing reducing the problems involved in the average intensity differences between the two views. Data normalization does not produce more reliable results.

The following step is the two images matching, on a defined disparity range. This calculation is applied by a crosscorrelation by shifting windows. The used criterion is the difference absolute value sum (DAS). We sort then the values to seek their minimum.

(1) Acquisition of left and right images.
(2) Rectification of left and right images.
(3) Local centering of left and right images.
(4) Matching using crosscorrelation.
  (a) Crosscorrelation computation (2).
  (b) Disparity computation, using the search of maximum value of crosscorrelation and subpixel interpolation.
(5) Filtering of the depth map.

Algorithm 1

Table 1: Operation required.

	Number of operations
Rectification	$2 \times 4 \times L \times H$
Local centering	$2 \times 21 \times L \times H$
Matching-crosscorrelation	$21 \times L \times H \times D$
Matching-max determination	$(2 \times D_r + 3) \times L \times H$
Total	$(23 \times D_r + 53) \times L \times H$

We evaluated the number of operations to be performed in order to map the algorithm on an embedded architecture.

In order to realize a fast processing of the local centering and the crosscorrelation, we use an optimized computing algorithm described hereafter. Because of this algorithm, the number of operations we carry out is no more proportional to the crosscorrelation window size.

So, we have to compute the following values:

$$C_r(x) = C_r(x-1) - C(x-l) + C(x),$$
$$C_{rc}(x) = C_{rc}(x-1) + C_r(x) - C_r(x-hL), \quad (9)$$

where, C_r and C_{rc} are intermediate values, x represents the current computed value index and $x-1$ the previous index, h and l are the height and the width of the crosscorrelation window and L is the image width. The C_r and C_{rc} values are the results of a previous computing of the crosscorrelation. C_r is the value computed in an h pixels row wide, and C_{rc} is the value computed in an $h \times l$ pixels window. These values must be computed in real-time in order not to break the data flow.

The C_r and C_{rc} values are 16 bits coded and must be stored into arrays. The capacities of these needed arrays are for C_{rc}, the line width, and for C_r, the line width multiplied by the crosscorrelation window height. This processing is therefore a more important consumer of memory space than a crosscorrelation classical computing. Moreover, memories must be managed with a lot of consideration in order not to break the data flow of the whole processing.

We examine in Table 1 the number of operations we need to realize the different processing. The operations we use are elementary and include simple arithmetic operations

Table 2: Virtex devices.

Device	System gates	CLB array	Logic cells	Block RAM bits	Block RAM number
XCV300	322970	32×48	6912	65536	16
XCV800	888439	56×84	21168	114688	28

(addition or subtraction), incrementations of values for the loops and access memory operations.

In this table, H and L are the height and the width of the image and D_r the disparity range value. After some studies of this algorithm working on human faces [23], we determined the optimal values for the crosscorrelation window size and the disparity range. We use a 256×256 pixels image size, a 7×6 pixels crosscorrelation window size and 20 for the disparity range. The number of operations we have to compute is then equal to 33, 62 Mops per trame, or 840 Mops per second for a 25-image-per-second video standard.

In order to optimize the implementation of the steps 1, 2, 3, and 4a in Algorithm 1 using parallel computing, we choose to use a reconfigurable logical device.

These processings are carried out effectively on the XILINX FPGA Virtex. Virtex devices provide better performance than previous generations of FPGA. Designs can achieve synchronous system clock rates up to 200 MHz including for Inputs-Outputs. Virtex devices feature a flexible regular architecture that comprises an array of configurable logic blocks (CLB) surrounded by programmable input/output blocks (IOBs), all interconnected by a hierarchy of fast and versatile routing resources. They incorporate also several large blocks RAM memories. Each block RAM is a fully synchronous dual-ported 4096-bit with independent control signal for each port. The data widths of the two ports can be configured independently. Thus, each block has 256 datas of 16- bit capacity. Each memory blocks are organized in columns. All Virtex devices contain two such columns, one along each vertical edge. The Virtex XCV300 and XCV800 capacities are grouped together in Table 2.

An original parallel implementation, described in the next paragraph, allows a very fast calculation of the criteria on all the disparity range.

These results are then given to a DSP which carries out successively the following processing: a parabolic interpolation to obtain wider disparity values; morphological filtering made up of an opening then a closing to eliminate wrong disparities while keeping depth map precision; a Gaussian blurring filter finally to smooth the obtained results. These processings are optimized on a C6x Texas Instrument DSP which allows a fast data processing.

3.1. Description of the chosen architecture

The constraints imposed by the algorithmic sequence real-time computing and the needed compactness to obtain an embedded architecture lead us to choose a reconfigurable and multiprocessor FPGA-DSP set: the Mirotech Arix Board. This board is designed in several independent computing parts, with configurable links. External links allow us to interface the board with a real-time Frame Grabber (FG) and with a PC (through the PCI bus). The computing parts are as follows (Figure 8): one virtual processing element (VPE) consisting of a Xilinx Virtex FPGA (XCV300 or XCV800) with four 512ko SBSRAM memory blocks; the second is composed of one Texas Instrument TMS320C44 DSP with two 1Mo SRAM memory blocks. This DSP interfaces two TIM sites on which we can connect the third computing element. For this part, we choose one Texas Instrument TMS320C67 DSP with an 8Mo SDRAM memory block. These three parts are connected by configurable links that allow direct memory access (DMA). Thus whole processing can be done in pipeline, cascaded in several parts as FG⇒VPE⇒DSPC44⇒DSPC67⇒DSPC44⇒PCI.

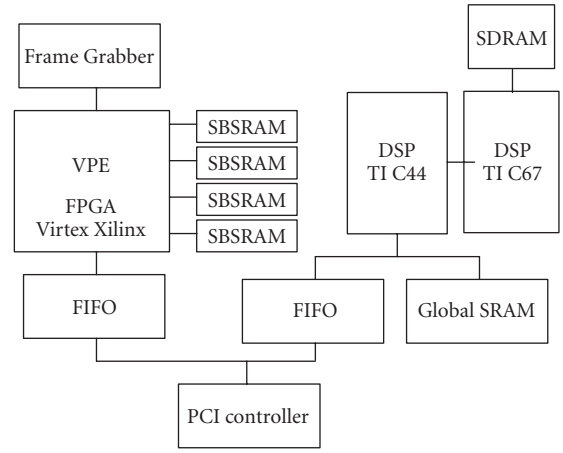


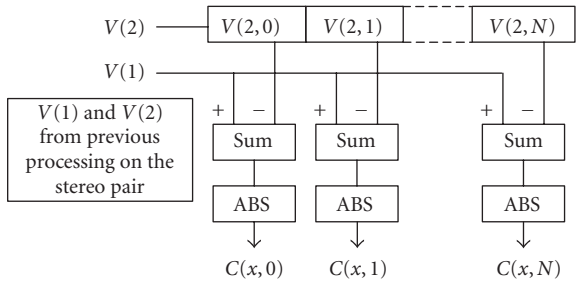
Figure 8: Parts of the board architecture.

This reconfigurable architecture allows us to quickly realize and validate our algorithm-architecture suitability.

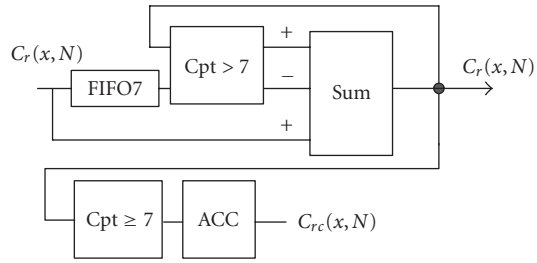
3.2. Matching implementation

The most important computation time is required by the matching processing; so we made a particular effort to implement this part. To obtain real-time results, we use the optimised crosscorrelation technique implemented using the intrinsic parallelism of FPGA.

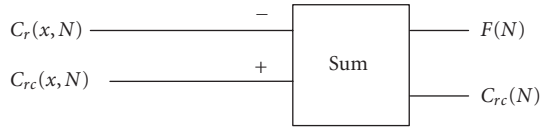
This method, described in a previous section, allows an important time gain by reusing intermediary computed results. Although a C language implementation of this algorithm is relatively simple, its FPGA implementation presents more problems. The main is memory management. Indeed, this processing needs a lot of intermediary values, easily allocated in C on a PC. Unfortunately, in order to respect the real-time constraint, we have to reduce the memory access and manage the best possible intermediary values and the data flow. Three processing parts are implemented: the first (Figure 9(a)) for the DAS parallel processing on the disparity



(a) DAS computing



(b) Intermediate values computing



(c) Final values computing

Figure 9: Matching implementation.

range; the second (Figure 9(b)), for the intermediary values parallel computing, and the third (Figure 9(c)) for the final computing. These two first parts hold, respectively, 9% and 6% slices of a Virtex300 FPGA.

For the parallel processing, we connect N times (N is between 0 and 19) the second part to all the outputs of the first part. We obtain thus in parallel the whole criteria needed to compute the disparity for one pixel. The C_{rc} criteria are stored in the Virtex memory blocks at the rate of one memory block per disparity. The C_r criteria are alternately stored into two SBSRAM blocks of the Arix board. For each even line, the writing is carried out into the first block and, for the odd lines, into the second block. The two memory blocks can then be used in parallel. This allows processing the third part, in which a reading of the C_{rc} and C_r criteria is carried out, without any influence onto the two other parts.

The whole final criteria, named $F(N)$, are then used for the determination of the maximum disparity onto the disparity range. The maximum disparity is determined, and we keep, with this value, the previous and the following disparity values. These three values are then sent to the DSP (which is well adapted to floating point processing) for a subpixel determination of the disparity (a parabolic interpolation, according (3)).

4. CONCLUSIONS AND PERSPECTIVES

We compared in the present paper various stereo matching methods in order to study real-time 3D face acquisition. We have shown that it is possible to implement a simple crosscorrelation-based algorithm with good performances, using post processing. A various architecture study led us to a judicious choice allowing obtaining the desired result. The real-time data processing is implemented on an embedded architecture. We obtain a dense face disparity map, precise enough for considered applications (multimedia, virtual worlds, biometrics) and using a reliable method. In particular, we plane to use the results as features for a face recognition software described in a previous article [26].

REFERENCES

[1] C. Beumier and M. Acheroy, "Automatic face verification from 3D and grey level clues," in *Proceedings of the 11th Portuguese Conference on Pattern Recognition (RECPAD '00)*, pp. 95–101, Porto, Portugal, May 2000.

[2] T. S. Jebara and A. Pentland, "Parametrized structure from motion for 3D adaptive feedback tracking of faces," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '97)*, pp. 144–150, San Juan, Puerto Rico, USA, June 1997.

[3] J. Y. Cartoux, *Formes dans les images de profondeur. Application à la reconnaissance et à l'authentification de visages*, Ph.D. thesis, Université Blaise Pascal, Clermont-Ferrand Cedex, France, 1989.

[4] M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of Cognitive Neuroscience*, vol. 3, no. 1, pp. 71–86, 1991.

[5] F. Devernay, *Vision stéréoscopique et propriétés différentielles des surfaces*, Ph.D. thesis, Ecole Polytechnique, l'Institut National de Recherche en Informatique et en Automatique, Chesnay Cedex, France, 1997.

[6] O. Faugeras, B. Hotz, H. Mathieu, et al., "Real time correlation based stereo: algorithm implementations and applications," Tech. Rep. RR-2013, l'Institut National de Recherche en Informatique et en Automatique, Chesnay Cedex, France, 1993.

[7] J.-R. Ohm and E. M. Izquierdo, "An object-based system for stereoscopic viewpoint synthesis," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 7, no. 5, pp. 801–811, 1997.

[8] B. Porr, A. Cozzi, and F. Wörgötter, "How to "hear" visual disparities: real-time stereoscopic spatial depth analysis using temporal resonance," *Biological Cybernetics*, vol. 78, no. 5, pp. 329–336, 1998.

[9] A. Koschan, "What is new in computational stereo since 1989: a survey of current stereo papers," Technischer Bericht 93-22, Technische Universiteät Berlin, Berlin, Germany, 1993.

[10] U. R. Dhond and J. K. Aggarwal, "Structure from stereo—a review," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 19, no. 6, pp. 1489–1510, 1989.

[11] L. Alvarez, R. Deriche, J. Sanchez, and J. Weickert, "Dense disparity map estimation respecting image discontinuities," Tech. Rep. 3874, l'Institut National de Recherche en Informatique et en Automatique, Chesnay Cedex, France, 2000.

[12] M. R. M. Jenkin and A. D. Jepson, "Recovering local surface structure through local phase difference measurements," *CVGIP: Image Understanding*, vol. 59, no. 1, pp. 72–93, 1994.

[13] P. Fua, "A parallel stereo algorithm that produces dense depth maps and preserves image features," *Machine Vision and Applications*, vol. 6, no. 1, pp. 35–49, 1993.

[14] R. Deriche and O. Faugeras, "Les EDP en traitement des images et vision par ordinateur," *Traitement du Signal*, vol. 13, no. 6, 1996.

[15] A. Fusiello, V. Roberto, and E. Trucco, "Efficient stereo with multiple windowing," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '97)*, pp. 858–863, San Juan, Puerto Rico, USA, June 1997.

[16] M. W. Maimone and S. A. Shafer, "Modeling foreshortening in stereo vision using local spatial frequency," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '95)*, vol. 1, pp. 519–524, Pittsburgh, Pa, USA, August 1995.

[17] J. Hoey, "Stereo disparity from local image phase," Tech. Rep., University of British Columbia, Vancouver, British Columbia, Canada, June 1999.

[18] I. Ohzawa, G. C. DeAngelis, and R. D. Freeman, "The neural coding of stereoscopic depth," *NeuroReport*, vol. 8, no. 3, pp. 3–12, 1997.

[19] P. Churchland and T. Sejnowski, *The Computational Brain*, MIT Press, Cambridge, Mass, USA, 1992.

[20] N. Kingsbury, "Image processing with complex wavelets," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 357, no. 1760, pp. 2543–2560, 1999, on a discussion meeting on "wavelets: the key to intermittent information?", London, UK, February 1999.

[21] H. Pan and J. Magarey, "Phase-based bidirectional stereo in coping with discontinuity and occlusion," in *Proceedings of International Workshop on Image Analysis and Information Fusion*, pp. 239–250, Adelaide, South Australia, November 1997.

[22] J. Magarey, A. Dick, P. Brooks, G. N. Newsam, and A. van den Hengel, "Incorporating the epipolar constraint into a multiresolution algorithm for stereo image matching," in *Proceedings of the 17th IASTED International Conference on Applied Informatics*, pp. 600–603, Innsbruck, Austria, February 1999.

[23] J.-P. Zimmer, "Modélisation de visage en temps réel par stéréovision," Thesis, University of Burgundy, Dijon, France, 2000.

[24] Z. Zhang, "Determining the epipolar geometry and its uncertainty: a review," *International Journal of Computer Vision*, vol. 27, no. 2, pp. 161–195, 1998.

[25] R. I. Hartley, "Theory and practice of projective rectification," *International Journal of Computer Vision*, vol. 35, no. 2, pp. 115–127, 1999.

[26] J.-P. Zimmer, J. Mitéran, F. Yang, and M. Paindavoine, "Security software using neural networks," in *Proceedings of the 24th Annual Conference of the IEEE Industrial Electronics Society (IECON '98)*, vol. 1, pp. 72–74, Aachen, Germany, August-September 1998.

Hindawi Publishing Corporation
EURASIP Journal on Image and Video Processing
Volume 2007, Article ID 38205, 11 pages
doi:10.1155/2007/38205

Research Article

Fusion of Appearance Image and Passive Stereo Depth Map for Face Recognition Based on the Bilateral 2DLDA

Jian-Gang Wang,[1] Hui Kong,[2] Eric Sung,[2] Wei-Yun Yau,[1] and Eam Khwang Teoh[2]

[1] *Institute for Infocomm Research, 21 Heng Mui Keng Terrace, Singapore 119613*
[2] *School of Electrical and Electronic Engineering, Nanyang Technological University, 50 Nanyang Avenue, Singapore 639798*

Received 27 April 2006; Revised 22 October 2006; Accepted 18 June 2007

Recommended by Christophe Garcia

This paper presents a novel approach for face recognition based on the fusion of the appearance and depth information at the match score level. We apply passive stereoscopy instead of active range scanning as popularly used by others. We show that present-day passive stereoscopy, though less robust and accurate, does make positive contribution to face recognition. By combining the appearance and disparity in a linear fashion, we verified experimentally that the combined results are noticeably better than those for each individual modality. We also propose an original learning method, the bilateral two-dimensional linear discriminant analysis (B2DLDA), to extract facial features of the appearance and disparity images. We compare B2DLDA with some existing 2DLDA methods on both XM2VTS database and our database. The results show that the B2DLDA can achieve better results than others.

1. INTRODUCTION

A great amount of research effort has been devoted to face recognition based on 2D face images [1]. However, the methods developed are sensitive to the changes in pose, illumination, and face expression. A robust identification system may require the fusion of several modalities because ambiguities in face recognition can be reduced with complementary multiple-modal information fusion. A multimodal identification system usually performs better than any one of its individual components, particularly in noisy environments [2]. One of the multimodal approaches is 2D plus 3D [3–7]. A good survey on 3D, 3D-plus-2D face recognition can be found in [8]. Intuitively, a 3D representation provides an added dimension to the useful information for the description of the face. This is because 3D information is relatively insensitive to change in illumination, skin-color, pose, and makeup; that is, it lacks the intrinsic weakness of 2D approaches. Studies [3–7, 9] have demonstrated the benefits of having this additional information. On the other hand, 2D image complements well 3D information. They are localized in hair, eyebrows, eyes, nose, mouth, facial hairs, and skin color precisely, where 3D capture is difficult and not accurate.

There are three main techniques for 3D facial surface capture. The first is by passive stereo using at least two cameras to capture a facial image and using a computational matching method. The second is based on structured lighting, in which a pattern is projected on a face and the 3D facial surface is calculated. Finally, the third is based on the use of laser range finding systems to capture the 3D facial surface. The third technique has the best reliability and resolution while the first has relatively poor robustness and accuracy. The attraction of passive stereoscopy is in its nonintrusive nature which is important in many real-life applications. Moreover, it is low cost. This serves as our motivation to use passive stereovision as one of the modalities of fusion and to ascertain if it can be sufficiently useful in face recognition. Our experiments, to be described later, will justify its use.

Currently, the 3D facial surface data quality obtained from the above three techniques is not comparable to that of the 2D images from a digital camera. The reason is that the 3D data usually have missing data or voids in the concave area of a surface, eyes, nostrils, and areas with facial hair. These issues are not problematic to an image from a digital camera. The facial surface data available to us from the XM2VTS database is also coarse (∼4000 points) compared to a 2D image (3 to 8 million pixels) from a digital camera and also compared to other 3D studies [3, 4], where they had around 200 000 points on the facial surface area. The cost of a 3D scanner is also much higher compared to a digital camera for taking 2D images.

While a lot of work has been carried out in face modeling and recognition, 3D information is still not widely used for recognition [10–12]. Initial studies concentrated on curvature analysis [13–15]. The existing 3D face recognition techniques proposed [10, 11, 16–22] assume the use of active 3D measurement for 3D face image capture. However, active methods employ structured illumination (structure projection, phase shift, etc.) or laser scanning, which is not desirable in many applications. Thanks to the technical progress in 3D capture/computing, an affordable real-time passive stereo system has become available. In this paper, we set out to find out if present-day passive stereovision in combination with 2D appearance images can match up to other methods relying on active depth data. Our main objective is to propose a method of combining appearance and depth face images to improve the recognition rate. While 3D face recognition research dates back to before 1990, algorithms that combine results from 3D and 2D data did not appear until about 2000 [17]. Pan et al. [23] used the Hausdorff distance for feature alignment and matching for 3D recognition. Recently, Chang et al. [3, 4, 16] applied principal components analysis (PCA) with 3D range data along with 2D image for face recognition. A Minolta Vivid 900 range scanner was used to obtain 2D and 3D images. Chang et al. [16] investigated the comparison and combination of 2D, 3D, and IR data for face recognition based on PCA representations of the face images. We note that their 3D data were captured by active scanning. Tsalakanidou [5] developed a system to verify the improvement of the face recognition rate by fusing depth and color eigenfaces on the XM2VTS database. The 3D models in the XM2VTS database are built using an active stereo system provided by the Turing Institute [24]. It can be seen that the recognition performance has been improved by using 3D information from the mentioned literature.

PCA and Fisher linear discriminant analysis (LDA) are common tools for facial feature extraction and dimension reduction. They have been successfully applied to face feature extraction and recognition [1]. The conventional LDA is a 1D feature extraction technique, and so a 2D image must first be vectorised before the application of LDA. Since the resulting image vectors are high-dimensional, LDA usually encounters the small sample size (SSS) problem in which the within-class scatter matrix becomes singular. Liu et al. [25] substituted $S_t = S_w + S_b$ for S_b to overcome the singularity problem. Yang et al. [26] proposed a 2DPCA for face recognition. Recently, some 2DLDA methods have been published [27–30] to solve SSS problem. In contrast to the $\mathbf{S}_b$ and $\mathbf{S}_w$ of 1DLDA, the corresponding $\mathbf{S}_b$ and $\mathbf{S}_w$ obtained by 2DLDA are not singular. Ye et al. [27] developed a scheme of simultaneous bilateral projections, L and R, and an iteration process to solve the two optimal projection metrics. This simultaneous bilateral projection is essentially a reprojection of a body of discriminant features that will discard some information. The performance of Ye's method depends on the initial choices of the transform matrix, R_0, and may lead to a local optimal solution although they suggested an initial R_0 based on their experiments. The focus of Ye's method is on the reduction of computational complexity of the conventional LDA method. Comparing with the conventional Fisherfaces (PCA plus LDA), Ye et al. found that the improvement in recognition accuracy by their 2DLDA method is not significant [27]. Yang et al. [29] and Visani et al. [30] developed a similar 2DLDA. These methods applied LDA in horizontal direction, and then applied LDA on the final left-projected features. This reprojection, however, may discard some discriminant information.

We proposed a novel 2DLDA framework containing unilateral 2DLDA (U2DLDA) and bilateral 2DLDA (B2DLDA) to overcome the SSS problem [28]. In this paper, we adopt the B2DLDA to extract facial features of the appearance and disparity images. Face is recognized by combining the appearance and disparity in a linear fashion. Differing from the existing 2DLDA [27, 29, 30], the B2DLDA keeps more discriminant information because the two sets of optimal discriminant features, which are obtained from either step of the asynchronous bilateral projection, are combined together for classification. We have compared our method to Ye's method in this paper. It shows better performance than Ye's 2DLDA because of the larger amount of discriminant information. In this paper, we also extended our work in [28] by comparing it with the existing 2DLDA approaches on stereo face recognition.

2. STEREO FACE RECOGNITION

So far, the reported 3D face recognition [3, 10, 16, 17] is based on active sensor (structure light, laser), however, they are not desirable in many applications. In this paper, we used SRI stereo engine [31] that outputs a high enough range resolution ($\leq$0.33 mm) for our applications. Our objective is to combine appearance and depth face images to improve the recognition rate. The performance of such fusion was evaluated on the commonly used database XM2VTS [32] and our own database collected by the real-time passive stereo vision system (SRI stereo engine, Mega-D [31]). The evaluation compares the results from appearance alone, depth alone, and the fusion of them, respectively. The performance using fused appearance and depth is the best among the three tests with a marked improvement of 5–8% accuracy. This justifies our method of fusion and also confirms our hypothesis that both modalities contribute positively. In Sections 2.1 and 2.2, we will discuss the generation of the 3D information of the XM2VTS and a passive stereo vision system. In Section 2.3, we will discuss the normalization of the 2D and 3D.

2.1. XM2VTS database

The XM2VTS is a large multimodal database. The faces are captured onto a high-quality digital video. It contains recordings of 295 subjects taken over a period of four months. Each recording contains a speaking head shot and a rotating head shot. Besides the digital video, the database provides high-quality color images, 32 KHz 16-bit sound files, and a 3D model, which deals with access control by the use of multimodal identification of human faces. The goal of using a multimodal recognition scheme is to improve the recognition efficiency by combining single modalities. We adopted

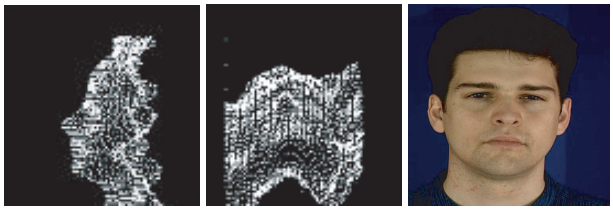


Figure 1: VRML model of a person's face.

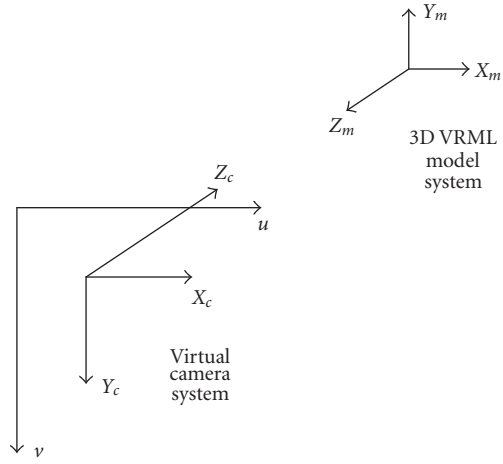


Figure 2: Geometric relationships among the virtual camera, 3D VRML model, and the image plane.

this database because 3D VRML models of subjects are provided and they can be used to generate the depth map for our algorithm. The high-precision 3D model of the subjects' head was built using an active stereo system provided by the Turing Institute [24]. In the following, we will discuss the generation of depth images from VRML model in the XM2VTS database.

A depth image is an image where the intensity of a pixel represents the depth of the correspondent point with respect to the 3D VRML model coordinate system. A 3D VRML model which contains the 3D coordinates and texture of a face in the XM2VTS database is displayed in Figure 1. There are about 4000 points in the 3D face model to represent the face. The face surface is triangulated with these points. In order to generate a depth image, a virtual camera is put in front of the 3D VRML model (Figure 2). The coordinate system of the virtual camera is defined as follows: the image plane is defined as the X-Y plane, the Z-axis is along the optical axis of the camera and pointing toward the frontal object. The camera plane, Y_c-Z_c, is positioned parallel to Y_m-X_m plane of the 3D VRML model. The Z_c coordinate aligns with Z_m coordinate, but in the reverse direction. X_c is antiparallel to X_m and Y_c is antiparallel to Y_m.

The intrinsic parameters of the camera must be properly defined in order to generate a depth image from a 3D VRML model. The parameters include (u_0, v_0), the coordinates of the image-center point (principle point); f_u and f_v, the scale factors of the camera along the u-axis and v-axis, respectively. The origin of the camera system under the 3D VRML model coordinate system is also set at (x_0, y_0, z_0).

The perspective projection pin-hole camera model is assumed. This means that for a point $F(x_m, y_m, z_m)$ in a 3D VRML model of a subject, the 2D coordinates of F in its depth image are computed as follows:

$$u = u_0 + \frac{f_u x_m}{z_0 - z_m},$$
$$v = v_0 - \frac{f_v y_m}{z_0 - z_m}. \tag{1}$$

In our approach, the z-buffering algorithm [33] is applied to handle the face self-occlusion for generating the depth images.

In the XM2VTS database, there is only one 3D model for each subject. In order to generate more than one view for learning and testing, some new views are obtained by rotating the 3D coordinates of the VRML model away from the frontal (about the Y_m axes) by some degrees. In our experiments, the new views are obtained at $\pm 3°$, $\pm 6°$, $\pm 9°$, $\pm 12°$, $\pm 15°$, $\pm 18°$.

2.2. Database collected by Mega-D

Here, we had used the SRI stereo head [31], in which the stereo process interpolates disparities up to 1/16 pixels. The resolution of the SRI stereo cameras is 640×480. Both intrinsic and extrinsic parameters are calibrated by an automatic calibration procedure. The smallest disparity change, Δd, is $(1/16) \times 7.5\,\mu\text{m} = 0.46875\,\mu\text{m}$. Here a pixel size of $7.5\,\mu\text{m}$. We used the Mega-D stereo head, where the baseline, b, is 9 cm and the focus length, f, is 16 mm. Hence when the distance from the subject to the stereo head, r, is 1 m, the range resolution, namely the smallest change in range that is discernable by the stereo geometry, is

$$\begin{aligned} \Delta r &= \left(\frac{r^2}{bf}\right)\Delta d \\ &= (1\,\text{m}^2/(90\,\text{mm} \times 16\,\text{mm})) \times 0.46875\,\mu\text{m} * 10^{-3} \\ &\approx 0.33\,\text{mm}. \end{aligned} \tag{2}$$

The range resolution is high enough for our face recognition applications. The manual of the SRI Small Vision System can be found in [31].

A database, called the Mega-D database, is collected using the SRI stereo head. The Mega-D database includes the images of 106 staff and students of our institute, with 12 pairs of appearance and disparity images for each subject. Two pairs per person are randomly selected for training while the remaining ten pairs are for testing. The recognition rate is calculated as the mean result of the experiments on these groups.

2.3. Normalizations of appearance and disparity images

Normalization is necessary to prevent the failure of similar face images of different sizes of the same person to be

recognised. The normalization of an appearance image of the XM2VTS or the Mega-D database is as follows: the appearance image is rotated and scaled to occupy a fixed size array of pixels using the image coordinates of the outer corners of the two eyes. The eye corners are extracted by our morphologically based method [34] and should be horizontal in the normalized images.

The normalization of a depth image in the XM2VTS database is as follows. The z values of the all pixels in the image are subtracted by a value in order that the distances between the nose tip and the camera are the same for all images.

In order to normalize a disparity image in the Mega-D database, we need to detect the outer corners of the two eyes and the nose tip in the disparity image. In the SRI stereo head, the coordinates of a pixel in the disparity image are consistent with the coordinates of the pixel in the left appearance image. Hence we can (more easily) detect the outer eye corners in the left appearance image instead of in the disparity image. The tip of the nose can be detected in the disparity image using template matching [11]. From the coplanar stereo vision model, we have

$$D = \frac{bf}{d}, \tag{3}$$

where D represents the depth, d is the disparity, b is the baseline, and f is the focal length of the calibrated stereo camera. The parameters b and f can be calibrated by the small vision system automatically. Hence we can get the depth image of a disparity image with (3). Thereby the depth image is normalised, similar to that in the XM2VTS database, using the depth of the nose tip. After that, the depth image is further normalized similarly by the outer corners of the two eyes.

In our approach, the normalized color images are changed to the gray-level image by averaging three channels:

$$I = \frac{R+G+B}{3}. \tag{4}$$

The parameters in (1) are set as

$$\begin{aligned} u_0 &= v_0 = 0, \\ f_x &= f_y = 4500, \\ x_0 &= y_0 = 0, \\ z_0 &= 20. \end{aligned} \tag{5}$$

Problems with the 3D data are alleviated to some degree by a preprocessing step to fill in holes (a region where there is missing 3D data during sensing) and spikes. We remove the holes by a median filter followed by linear interpolation of missing values from good values around the edges of the holes.

Some of the normalized face image samples in the XM2VTS database are shown in Figure 3, where color face images are shown in Figure 3(a) and the corresponding depth images are shown in Figure 3(b). The size of the normalized image is 88×64. We can see significant changes in illumination, expressions, hair, and eye glasses/no eyeglasses due to longer time lapse (four months) in photograph taking.

Samples of the normalized face images in the Mega-D database are shown in Figures 4 and 5. Both color face images and the corresponding disparity images are shown in Figure 4. The resolution of the images is 88×64. The distance between the subjects and the camera is about 1.5 m. We can see some changes in illumination, pose, and expression in Figure 5.

3. FEATURE EXTRACTION

We have proposed a bilateral two-dimensional linear discriminant analysis (B2DLDA) [28] to solve the small sample size problem. In this paper, we apply it to extract features of appearance and depth images. Here, we will extend the work in [28] by comparing it with existing 2DLDA approaches [27, 29, 30].

3.1. B2DLDA algorithm

The pseudocode for the B2DLDA algorithm is given in Algorithm 1.

For face classification, $\mathbf{W}_l$ and $\mathbf{W}_r$ are applied to a probe image to obtain the features B_l and B_r. The B_l and B_r are converted to 1D vector, respectively. PCA is adopted to classify the concatenated vectors of $\{B_l, B_r\}$. It is noted that PCA or LDA can be used in this step. Ye et al. [27] adopted LDA to reduce the dimension of 2DLDA, since a small reduced dimension is desirable for efficient querying. We used PCA because we try to keep as much structure of the features (variance). There are at most $C - 1$ discriminant components corresponding to nonzero eigenvalues. Their numbers, m_l and m_r, can be selected using the Wilks Lambda criteria, which is known as the stepwise discriminant analysis [35]. This analysis shows that the number of discriminant components required by left and right transforms for our case is 20. So for our experiments, we set $m_l = m_r = 20$. We used the same number of principal components for classification. This choice was verified experimentally as using more than 20 discriminant components did not improve the results.

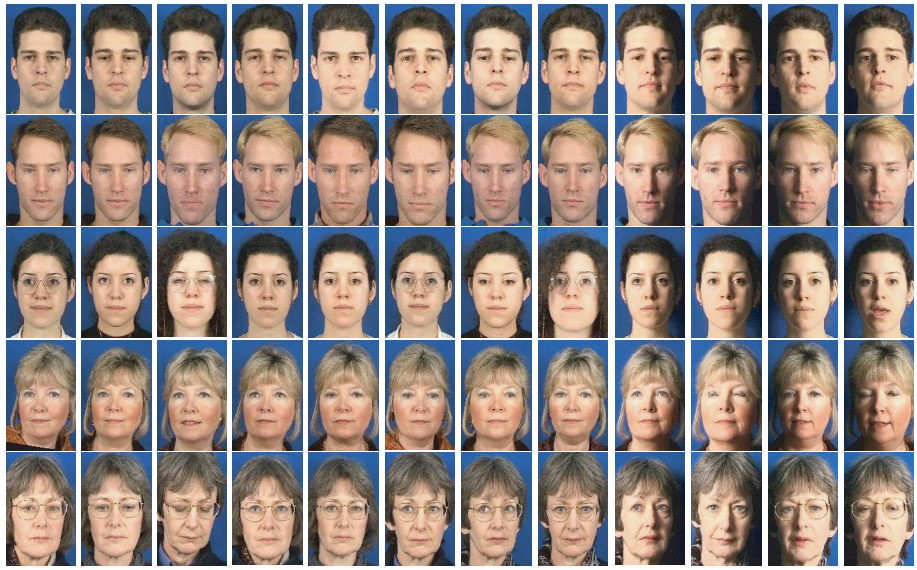
3.2. The complexity analysis

We can see that the most expensive steps in Algorithm 1 are in lines 3, 6, 9. The comparisons of computational complexity of Fisherfaces, Ye's 2DLDA, Yang's 2DLDA, and the proposed 2DLDA are listed in Table 1.

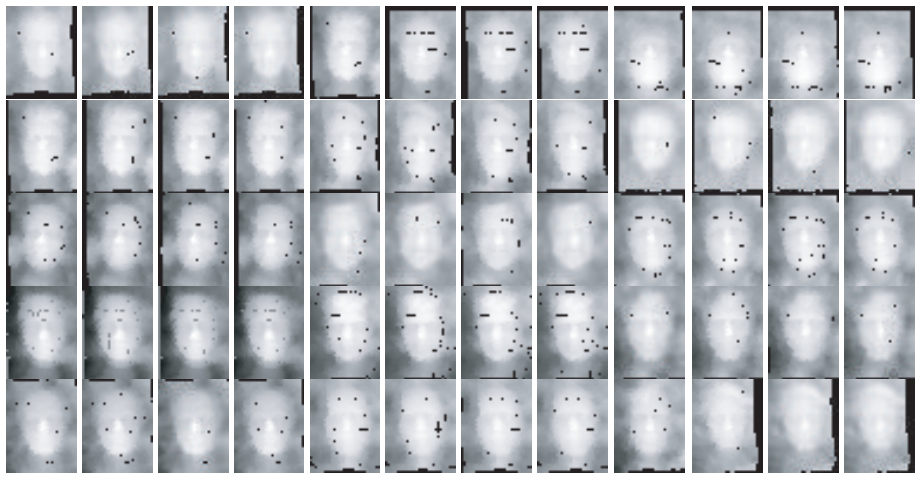
The computational complexity of Fisherfaces increases cubically with the size of the training sample size. The computational complexity of B2LDA is the same as Yang's method, and both of them depend on the image size. However, it is higher than Ye's method.

4. FUSION OF APPEARANCE AND DEPTH/DISPARITY

We aim to improve the recognition rate by combining appearance and depth information. The matter of how to fuse two or more sources of information is crucial to the



(a) Normalized color face images: columns 1–4: images in CDS001; columns 5–8: images in CDS006; columns 9–12: images in CDS008



(b) Normalized depth images corresponding to (a)

Figure 3: Normalized 2D and 3D face images in the XM2VTS database: (a) appearance images, (b) depth images.

performance of the system. The criterion for this kind of combination is to fully make use of the advantages of the two sources of information to optimize the discriminant power of the whole system. The degree to which the results improve performance is dependent on the degree of correlation among individual decisions. Fusion of decisions with low mutual correlation can dramatically improve the performance. There is a rich literature [2, 36] on fusing multiple modals for identity verification, for example, combining voice and fingerprints, voice and face biometrics [37], and visible and thermal imagery [38]. The fusion can be done at the feature level, matching score level, or decision level. In this paper, we are interested in the fusion at the matching score level. There are some ways of combining different matching scores to achieve the best decision, for example, by majority vote, sum rule, multiplication rule, median rule, minimum rule, and average rule. It is known that sum and multiplication rules provide general plausible results. In this paper, we use the weighted sum rule to fuse appearance and depth information. Our rationale is that appearance information and depth information are quite highly uncorrelated. This is clear since depth data yields surface or terrain of the observed scene while the appearance information records the texture of the surface. Though the normals to the surface affects the reflectivity of light and thereby the surface illumination, this has minimal effect on the surface texture. Therefore, a certain linear combination will be sufficient to extract a good set of features for the purpose of recognition. Nevertheless, there will be a small correlation between them in the sense that the general terrain of the face (i.e., depth map) has

Figure 4: Normalized appearance and disparity images captured by the Mega-D stereo head.

Figure 5: Normalized appearance images captured by a Mega-D stereo head.

a bearing on the shading of the appearance image. We investigate the complete range of linear combinations to reveal the interplay between these two paradigms.

The linear combination of the appearance and depth in our approach can be explained using Figure 6. We optimize the combination of the depth and intensity discriminant Euclidean distances by minimizing the weighted sum of two discriminant Euclidean distances.

Given the gallery of depth images and appearance images, they are trained, respectively, by B2DLDA. The Euclidean distance between the test image and the templates are measured as the inverse of similarity score to decide whose face it is. Assuming the eigenvectors of face image k and i are represented as $\mathbf{v}_k$ and $\mathbf{v}_i$, respectively,

$$S^{-1}(k,i) = \text{dist}(k,i) = ||\mathbf{v}_k - \mathbf{v}_i||_2. \tag{6}$$

A probe face, F_T, is identified as a face, F_L, of the gallery if the sum of the weighted similarity scores (appearance and depth) from F_T to F_L is the maximum among such sums from F_T to all the faces in the gallery. This can be expressed as

$$\max_{\text{gallery}} \{w_1 S_{2D} + (1 - w_1) S_{3D}\}, \tag{7}$$

where S_{2D} and S_{3D} are the similarity scores for intensity and depth images, respectively. The weight w_1 is determined to be optimal through experiments. In general, a higher value of $(1 - w_1)$ reflects the fact that the variance of the discriminant Euclidean distance of a depth map is relatively smaller than the one for the corresponding appearance face image.

5. EXPERIMENTAL RESULTS

The face recognition experiments are performed on the XM2VTS database and the Mega-D database, respectively, to verify the improvement of the recognition rate by combining 2D and 3D information. We assess the accuracy and efficiency of B2DLDA and compare it with Ye's 2DLDA [27], Yang's 2DLDA [29], Fisherfaces [34], and Eigenfaces [3–5].

Input: $A_1, A_2, \ldots, A_n, m_l, m_r$ % A_i are the n images, and m_l and m_r are the number of the % discriminant components of left and right B2DLDA transform

Output: $\mathbf{W}_l, \mathbf{W}_r, B_{l1}, B_{l2}, \ldots, B_{ln}, B_{r1}, B_{r2}, \ldots, B_{rn}$ % $\mathbf{W}_l$ and $\mathbf{W}_r$ are the left and right % transformation matrix respectively by % B2DLDA; B_{li} and B_{ri} are the reduced % representations of A_i by $\mathbf{W}_l$ and $\mathbf{W}_r$ % respectively

(1) Compute the mean, $\mathbf{M}_i$, of the ith class of each i

(2) Compute the global mean, $\mathbf{M}$, of $\{A_i\}$, $i = 1, 2, \ldots, n$

(3) Find $\mathbf{S}_{bl}$ and $\mathbf{S}_{wl}$, $\mathbf{S}_{bl} = \sum_{i=1}^{C} C_i \bullet (\mathbf{M}_i - \mathbf{M})^T (\mathbf{M}_i - \mathbf{M})$, $\mathbf{S}_{wl} = \sum_{i=1}^{C} \sum_{j=1}^{C_i} (\mathbf{X}_i^j - \mathbf{M_i})^T (\mathbf{X}_i^j - \mathbf{M}_i)$ % C is the number of the classes; C_i is the % number of the samples in the ith class

(4) Compute the first m_l eigenvectors $\{\phi_i^L\}_{i=1}^{m_l}$ of $\mathbf{S}_{\mathbf{wl}}^{-\mathbf{1}} \mathbf{S}_{\mathbf{b}l}$

(5) $\mathbf{W}_l \leftarrow [\phi_1^L, \phi_2^L, \ldots, \phi_{m_l}^L]$

(6) Find $\mathbf{S}_{br}$ and $\mathbf{S}_{wr}$, $\mathbf{S}_{br} = \sum_{i=1}^{C} C_i \bullet (\mathbf{M}_i - \mathbf{M})(\mathbf{M}_i - \mathbf{M})$, $\mathbf{S}_{wr} = \sum_{i=1}^{C} \sum_{j=1}^{C_i} (\mathbf{X}_i^j - \mathbf{M}_i)(\mathbf{X}_i^j - \mathbf{M})$

(7) Compute the first m eigenvectors $\{\phi_i^R\}_{i=1}^{m_r}$ of $\mathbf{S}_{\mathbf{wr}}^{-\mathbf{1}} \mathbf{S}_{\mathbf{b}r}$

(8) $\mathbf{W}_r \leftarrow [\phi_1^R, \phi_2^R, \ldots, \phi_{m_r}^R]$

(9) $B_{li} = A_i W_l$, $i = 1, \ldots, n$

$B_{ri} = A_i' W_r$, $i = 1, \ldots, n$

(10) Return $\mathbf{W}_l, \mathbf{W}_r, B_{li}, B_{ri}$, $i = 1, \ldots, n$

Algorithm 1: Algorithm B2DLDA $(\mathbf{A}_1, \mathbf{A}_2, \ldots, \mathbf{A}_n, m_l, m_r)$.

Table 1: The comparisons of computational complexity of Fisherfaces [39], Ye's 2DLDA [27], Yang's 2D LDA [29], and the proposed 2DLDA [28]. M is the total number of the train samples; r, c are the numbers of the rows and columns of the original image, $\mathbf{A}$, respectively; $l = \max(r, c)$.

Method	Fisherfaces [39]	Ye [27]	Yang [29]	B2DLDA [28]
Computation complexity	$O(M^3)$	$O(rc)$	$O(l^3)$	$O(l^3)$

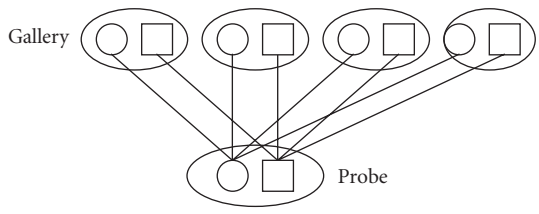


Figure 6: Combination of appearance (circle) and depth (square) information.

5.1. Experiment on the XM2VTS database

The XM2VTS consists of the frontal and profile views of 295 subjects. We used the frontal views in the XM2VTS database (CDS001, CDS006, and CDS008 darkened frontal view). CDS001 dataset contains one frontal view for each of the 295 subjects and each of the four sessions. This image was taken at the beginning of the head rotation shot. So there are a total of 1180 color images, each with a resolution of 720×576 pixels. CDS006 dataset contains one frontal view for each of the 295 subjects and each of the four sessions. This image was taken from the middle of the head rotation shot when the subject had returned his/her head to the middle. They are different from those contained in CDS001. There are a total of 1180 color images. The images are at a resolution of 720×576 pixels. CDS008 contains four frontal views for each of the 295 subjects taken from the final session. In two of the images, the studio light illuminating the left side of the face was turned off. In the other two images, the light illuminating the right side of the face was turned off. There are a total of 1180 color images. The images are at a resolution of 720×576 pixels. We used the 3D VRML model (CDS005) of the XM2VTSDB to generate 3D depth images corresponding to the appearance images mentioned above. The models were obtained with a high-precision 3D stereo camera developed by the Turing Institute [24]. The models were then converted from their proprietary format into VRML.

Therefore, a total of 3540 pairs of frontal views (appearance and depth pair) of 295 subjects in X2MVTS database are used. There are 12 pairs of images for each subject. We pick randomly any two of them for the learning gallery while the remainder ten pairs per subject are used as probes. The average recognition rate was obtained over 66 random runs. As only two pairs of face images are used for training, it is clear that LDA will face the SSS problem because the number of the training samples is much less than the dimension of the covariance matrix in LDA. Using two images per person for training could be insufficient for LDA-based or

Table 2: The mean recognition rates (%) on the XM2VTS database versus w_1.

w_1	B2DFDA [28]	Ye's 2D LDA [27]	Yang's 2DLDA [29]	Fisherfaces [39]	Eigenfaces [3–5]
0.0	91.63	90.88	89.88	87.86	84.86
0.1	97.88	96.00	95.00	94.80	93.10
0.2	98.66	97.44	96.44	96.10	94.50
0.3	97.88	96.66	95.66	95.20	92.52
0.4	97.81	96.01	95.01	94.80	91.80
0.5	95.75	94.38	93.92	93.81	90.90
0.6	94.19	93.61	93.01	92.80	90.14
0.7	94.19	93.14	92.14	91.40	89.40
0.8	91.84	91.58	90.58	88.50	87.51
0.9	88.72	88.84	87.84	86.90	85.90
1.0	81.69	80.63	78.63	76.70	75.71

Table 3: The mean recognition rates (%) on the Mega-D database versus w_1.

w_1	B2DFDA [28]	Ye's 2D LDA [27]	Yang's 2DLDA [29]	Fisherfaces [39]	Eigenfaces [3–5]
0.0	90.63	89.87	88.78	89.80	83.82
0.1	97.56	95.44	94.41	94.17	92.51
0.2	96.88	95.00	94.02	93.78	92.13
0.3	96.82	94.62	93.60	93.23	90.51
0.4	95.31	94.01	93.04	92.81	89.78
0.5	93.73	92.81	92.92	90.84	88.92
0.6	92.18	92.01	92.00	90.30	88.17
0.7	92.10	91.14	90.03	88.39	87.41
0.8	89.83	89.60	88.42	86.49	85.53
0.9	86.71	86.79	85.70	85.91	83.91
1.0	79.69	78.58	74.61	78.72	73.73

2DLDA-based face recognition to be optimal. In this paper, we want to show that our proposed method can solve the SSS problem where the number of training sample is less. Therefore, we used the least images per person, that is two, for training. It is fair to compare our algorithm with others because we used the same training set for this comparison. Thus our algorithm is useful in situations where there are only limited numbers of samples for training.

Using the training gallery and probe described above, the evaluations of the recognition algorithms on B2DLDA, Ye's 2DLDA, Yang's 2DLDA, Fisherfaces, and eigenfaces have been done. This includes the recognition evaluation when the weight w_1 in (7) is varied from 0 (which corresponds to depth alone) to 1 (which corresponds to intensity alone) with a step increment of 0.1. Assuming we have N training samples of C subjects (classes), the recognition rates on the XM2VTS database versus the weight w_1 are given in Table 2 or Figure 7. B2DFDA is compared with

(1) Ye's 2D LDA [27],

(2) Yang's 2DLDA [29],

(3) Fisherfaces (PCA plus LDA) [39],

(4) Eigenfaces [3–5].

By fusing the appearance and the depth, the highest recognition rate, 98.66%, happens at $w_1 = 0.2$ for B2DLDA as shown in Table 2. This supports our hypothesis that the combined method outperforms the individual appearance or depth. The results in Table 2 also verified that the proposed B2DLDA outperforms Ye's 2DLDA. Ye reported their method can get the results similar to optimal LDA (PCA + LDA). Here, this can be observed in our results.

5.2. Experiment on stereo vision system

Differing from the existing 3D or 2D + 3D face recognition systems, we used a passive stereovision to get 3D information. A database, called Mega-D, was built with SRI stereo head engine. (We have described the Mega-D database in Section 3.2.) In this section, we evaluate the algorithms on the Mega-D database. We will show that we can get comparable results with the database where 3D information is obtained by an active stereo engine, that is, the XM2VTS database.

A total of 1272 frontal views of 106 subjects in the Mega-D database are used. There are 12 pairs of images for each subject. We use any two randomly selected pairs of them

Table 4: The computation time of Fisherfaces [39], Ye's 2DLDA [27], Yang's 2DLDA [29], and the proposed 2DLDA [28].

Method	Fisherfaces [39]	Ye's 2DLDA [27]	Yang's 2DLDA [29]	B2DLDA [28]
CPU time (s)	75	12.5	24	26

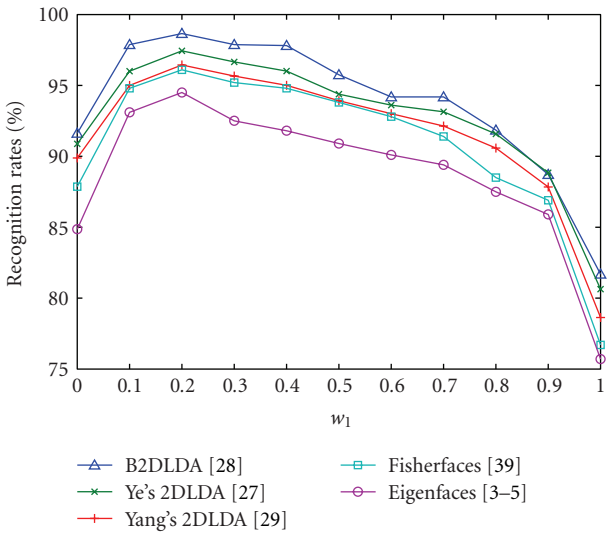


Figure 7: Recognition performance on the XM2VTS database versus w_1. $w_1 = 0$ corresponds to 3D alone, $w_1 = 1$ corresponds to 2D alone.

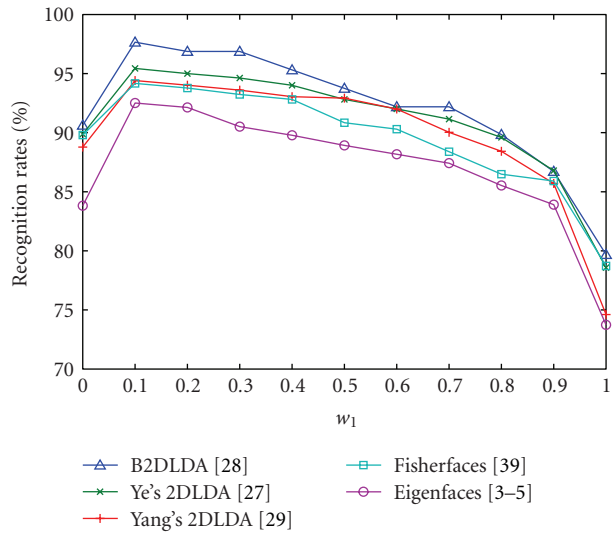


Figure 8: Recognition performance on the Maga-D database versus w_1. $w_1 = 0$ corresponds to 3D alone, $w_1 = 1$ corresponds to 2D alone.

for the learning gallery while the remainder ten are used as probes. Using the gallery and probe described above, the evaluations of the recognition algorithms (2D FDA and 1D FDA) have been done, include the recognition when the weight w_1 in (7) varies from 0 (which corresponds to depth alone) to 1 (which corresponds to intensity alone) with a step increment of 0.1. Similar to the experiments on the XM2VTS database, a total of 66 random trials were performed and the mean of these trails is used in the final recognition result. The recognition rates on the Mega-D database versus the weight w_1 are given in Table 3 or Figure 8.

Similar to the results on the XM2VTS database, the results supported our hypothesis that the combined method outperforms the individual appearance or depth. It also verified that the proposed B2DLDA outperforms Ye's 2DLDA. Ye's method [27] can get the results similar to Fisherfaces. This experiment also illustrated the viability of using passive stereovision for face recognition.

We implemented the algorithms in Visual C++ on a P3 3.4Ghz 1GB PC. The computation time is listed in Table 4.

We can see in Table 4 that our method's processing time costs twice more than that for Ye's method (only one iteration).

6. CONCLUSIONS

In this paper, a novel fusion of appearance image and passive stereo depth is proposed to improve face recognition rates. Different from the existing 3D or 2D + 3D face recognition that used active stereo method to obtain 3D information, comparable results have been obtained in this paper on both the XM2VTS and a large database collected with the passive Mega-D stereo engine. We investigated the complete range of linear combinations to reveal the interplay between these two paradigms. The improvement of the face recognition rate using this combination has been verified. The recognition rate by the combination is better than either appearance alone or depth alone. In order to overcome the small sample size problem in LDA, a bilateral two-dimensional linear discriminant analysis (B2DLDA) is proposed in this paper to extract the image features. The experimental results show that B2DLDA outperforms the existing 2DLDA approaches.

REFERENCES

[1] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recognition: a literature survey," *ACM Computing Surveys*, vol. 35, no. 4, pp. 399–458, 2003.

[2] R. Brunelli and D. Falavigna, "Person identification using multiple cues," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 10, pp. 955–966, 1995.

[3] K. Chang, K. Bowyer, and P. Flynn, "Face recognition using 2D and 3D facial data," in *Proceedings of ACM Workshop on Multimodal User Authentication*, pp. 25–32, Santa Barbara, Calif, USA, December 2003.

[4] K. I. Chang, K. W. Bowyer, and P. J. Flynn, "An evaluation of multimodal 2D+3D face biometrics," *IEEE Transactions on*

Pattern Analysis and Machine Intelligence, vol. 27, no. 4, pp. 619–624, 2005.

[5] F. Tsalakanidou, D. Tzovaras, and M. G. Strintzis, "Use of depth and colour eigenfaces for face recognition," *Pattern Recognition Letters*, vol. 24, no. 9-10, pp. 1427–1435, 2003.

[6] J.-G. Wang, H. Kong, and R. Venkateswarlu, "Improving face recognition performance by combining colour and depth fisherfaces," in *Proceedings of 6th Asian Conference on Computer Vision*, pp. 126–131, Jeju, Korea, January 2004.

[7] J.-G. Wang, K.-A. Toh, and R. Venkateswarlu, "Fusion of appearance and depth information for face recognition," in *Proceedings of the 5th International Conference on Audio- and Video-Based Biometric Person Authentication (AVBPA '05)*, pp. 919–928, Rye Brook, NY, USA, July 2005.

[8] K. W. Bowyer, K. Chang, and P. Flynn, "A survey of approaches and challenges in 3D and multi-modal 3D + 2D face recognition," *Computer Vision and Image Understanding*, vol. 101, no. 1, pp. 1–15, 2006.

[9] N. Mavridis, F. Tsalakanidou, D. Pantazis, S. Malassiotis, and M. G. Strintzis, "The HISCORE face recognition application: affordable desktop face recognition based on a novel 3D camera," in *Proceedings of International Conference on Augmented, Virtual Environments and Three Dimensional Imaging (ICAV3D '01)*, pp. 157–160, Mykonos, Greece, May-June 2001.

[10] C. Beumier and M. Acheroy, "Automatic face authentication from 3D surface," in *Proceedings of British Machine Vision Conference (BMVC '98)*, pp. 449–458, Southampton, UK, September 1998.

[11] G. G. Gordon, "Face recognition based on depth maps and surface curvature," in *Geometric Methods in Computer Vision*, vol. 1570 of *Proceedings of SPIE*, pp. 234–247, San Diego, Calif, USA, July 1991.

[12] X. Lu and A. K. Jain, "Deformation analysis for 3D face matching," in *Proceedings of the 7th IEEE Workshop on Applications of Computer Vision / IEEE Workshop on Motion and Video Computing (WACV/MOTION '05)*, pp. 99–104, Breckenridge, Colo, USA, January 2005.

[13] P. J. Philips, P. Grother, R. J. Micheals, D. M. Blackburn, E. Tabassi, and M. Bone, "Face recognition vendor test 2002," Tech. Rep. NIST IR 6965, National Institute of Standards and Technology, Gaithersburg, Md, USA, March 2003.

[14] S. A. Rizvi, P. J. Phillips, and H. Moon, "The FERET verification testing protocol for face recognition algorithms," Tech. Rep. NIST IR 6281, National Institute of Standards and Technology, Gaithersburg, Md, USA, October 1998.

[15] P. J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss, "The FERET evaluation methodology for face-recognition algorithms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 10, pp. 1090–1104, 2000.

[16] K. I. Chang, K. W. Bowyer, P. J. Flynn, and X. Chen, "Multibiometrics using facial appearance, shape and temperature," in *Proceedings of the 6th IEEE International Conference on Automatic Face and Gesture Recognition (FGR '04)*, pp. 43–48, Seoul, Korea, May 2004.

[17] C. Beumier and M. Acheroy, "Face verification from 3D and grey level clues," *Pattern Recognition Letters*, vol. 22, no. 12, pp. 1321–1329, 2001.

[18] J. C. Lee and E. E. Milios, "Matching range images of human faces," in *Proceedings of the 3rd International Conference on Computer Vision (ICCV '90)*, pp. 722–726, Osaka, Japan, December 1990.

[19] Y. Yacoob and L. S. Davis, "Labeling of human face components from range data," *CVGIP: Image Understanding*, vol. 60, no. 2, pp. 168–178, 1994.

[20] C.-S. Chua, F. Han, and Y. K. Ho, "3D human face recognition using point signature," in *Proceedings of the 4th IEEE International Conference on Automatic Face and Gesture Recognition (FG '00)*, pp. 233–238, Grenoble, France, March 2000.

[21] V. Blanz and T. Vetter, "Face recognition based on fitting a 3D morphable model," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 9, pp. 1063–1074, 2003.

[22] V. Blanz and T. Vetter, "A morphable model for the synthesis of 3D faces," in *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '99)*, pp. 187–194, Los Angeles, Calif, USA, August 1999.

[23] G. Pan, Y. Wu, and Z. Wu, "Investigating profile extracted from range data for 3D face recognition," in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, vol. 2, pp. 1396–1399, Washington, DC, USA, October 2003.

[24] C. W. Urquhart, J. P. McDonald, J. P. Siebert, and R. J. Fryer, "Active animate stereo vision," in *Proceedings of the 4th British Machine Vision Conference*, pp. 75–84, University of Surrey, Guildford, UK, September 1993.

[25] K. Liu, Y.-Q. Cheng, and J.-Y. Yang, "Algebraic feature extraction for image recognition based on an optimal discriminant criterion," *Pattern Recognition*, vol. 26, no. 6, pp. 903–911, 1993.

[26] J. Yang, D. Zhang, A. F. Frangi, and J.-Y. Yang, "Two-dimensional PCA: a new approach to appearance-based face representation and recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 1, pp. 131–137, 2004.

[27] J. Ye, R. Janardan, and Q. Li, "Two-dimensional linear discriminant analysis," in *Proceedings of Neural Information Processing Systems (NIPS '04)*, pp. 1569–1576, Vancouver, British Columbia, Canada, December 2004.

[28] H. Kong, L. Wang, E. K. Teoh, J.-G. Wang, and R. Venkateswarlu, "A framework of 2D fisher discriminant analysis: application to face recognition with small number of training samples," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, vol. 2, pp. 1083–1088, San Diego, Calif, USA, June 2005.

[29] J. Yang, D. Zhang, X. Yong, and J.-Y. Yang, "Two-dimensional discriminant transform for face recognition," *Pattern Recognition*, vol. 38, no. 7, pp. 1125–1129, 2005.

[30] M. Visani, C. Garcia, and J.-M. Jolion, "Two-dimensional-oriented linear discriminant analysis for face recognition," in *Proceedings of the International Conference on Computer Vision and Graphics (ICCVG '04)*, pp. 1008–1017, Warsaw, Poland, September 2004.

[31] Videre Design, "MEGA-D Megapixel Digital Stereo Head," http://users.rcn.com/mclaughl.dnai/sthmdcs.htm.

[32] K. Messer, J. Matas, J. Kittler, J. Luettin, and G. Maitre, "XM2VTSDB: the extended M2VTS database," in *Proceedings of International Conference on Audio- and Video-Based Biometric Person Authentication (AVBPA '99)*, pp. 72–77, Washington, DC, USA, March 1999.

[33] E. E. Catmull, *A subdivision algorithm for computer display of curved surfaces*, Ph.D. thesis, Department of Computer Science, University of Utah, Salt Lake City, Utah, USA, 1974.

[34] J.-G. Wang and E. Sung, "Frontal-view face detection and facial feature extraction using color and morphological operations," *Pattern Recognition Letters*, vol. 20, no. 10, pp. 1053–1068, 1999.

[35] R. I. Jenrich, "Stepwise discriminant analysis," in *Statistical Methods for Digital Computers*, K. Enslein, A. Ralston, and H.

S. Wilf, Eds., pp. 76–95, John Wiley & Sons, New York, NY, USA, 1977.

[36] J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas, "On combining classifiers," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 3, pp. 226–239, 1998.

[37] T. Choudhury, B. Clarkson, T. Jebara, and A. Pentland, "Multimodal person recognition using unconstrained audio and video," in *Proceedings of the 2nd International Conference on Audio- and Video-Based Person Authentication (AVBPA '99)*, pp. 176–181, Washington, DC, USA, March 1999.

[38] D. A. Socolinsky, A. Selinger, and J. D. Neuheisel, "Face recognition with visible and thermal infrared imagery," *Computer Vision and Image Understanding*, vol. 91, no. 1-2, pp. 72–114, 2003.

[39] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. fisherfaces: recognition using class specific linear projection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711–720, 1997.

Hindawi Publishing Corporation
EURASIP Journal on Image and Video Processing
Volume 2007, Article ID 17173, 8 pages
doi:10.1155/2007/17173

Research Article

Localized versus Locality-Preserving Subspace Projections for Face Recognition

Iulian B. Ciocoiu[1] and Hariton N. Costin[2, 3]

[1] *Faculty of Electronics and Telecommunications, "Gh. Asachi" Technical University of Iaşi, 700506 Iaşi, Romania*
[2] *Faculty of Medical Bioengineering, "Gr. T. Popa" University of Medicine and Pharmacy, 700115 Iaşi, Romania*
[3] *Institute for Theoretical Computer Science, Romanian Academy, Iaşi Branch, 700506 Iaşi, Romania*

Received 1 May 2006; Revised 10 September 2006; Accepted 26 March 2007

Recommended by Tim Cootes

Three different localized representation methods and a manifold learning approach to face recognition are compared in terms of recognition accuracy. The techniques under investigation are (a) local nonnegative matrix factorization (LNMF); (b) independent component analysis (ICA); (c) NMF with sparse constraints (NMFsc); (d) locality-preserving projections (Laplacian faces). A systematic comparative analysis is conducted in terms of distance metric used, number of selected features, and sources of variability on AR and Olivetti face databases. Results indicate that the relative ranking of the methods is highly task-dependent, and the performances vary significantly upon the distance metric used.

1. INTRODUCTION

Face recognition has represented for more than one decade one of the most active research areas in pattern recognition. A plethora of approaches has been proposed and evaluation standards have been defined, but current solutions still need to be improved in order to cope with the recognition rates and robustness requirements of commercial products. A number of recent surveys [1, 2] review modern trends in this area of research, including

(a) kernel-type extensions of classical linear subspace projection methods such as kernel PCA/LDA/ICA [3–6];

(b) holistic versus component-based approaches [7, 8], compared in terms of stability to local deformations, lighting variations, and partial occlusion. The list is augmented by representation procedures using space-localized basis images, three of which are described in the present paper;

(c) the assumption that many real-world data lying near low-dimensional nonlinear manifolds exhibiting specific structure triggered the use of a significant set of manifold learning strategies in face-oriented applications [9, 10], two of which are included in the present comparative analysis.

Recent publications have addressed many other important issues in still-face image processing, such as yielding robustness against most of the sources of variability, dealing with the small sample size problem, or automatic detection of fiducial points. Despite the continuously growing number of solutions reported in the literature, little has been done in order to make fair comparisons in terms of face recognition performances based on a unified measurement protocol and using realistic (large) databases. A remarkable exception is represented by the face recognition vendor test [11] conducted by the National Institute of Standards and Technology (NIST) since 2000 (following the widely known FERET evaluations), complemented by the face recognition grand challenge.

The present paper focuses on a systematic comparative analysis of subspace projection methods using localized basis functions, against techniques using locality-preserving constraints. We have conducted extensive computer experiments on AR and Olivetti face databases and the techniques under investigation are (a) local nonnegative matrix factorization (LNMF) [12]; (b) independent component analysis (ICA) [13]; (c) nonnegative Matrix Factorization with sparse constraints (NMFsc) [14]; and (d) locality-preserving projections (Laplacian faces) [9]. We have taken into account a number of design issues, such as the type of distance metric, the dimension of the feature vectors to be used for actual classification, and the sources of face variability.

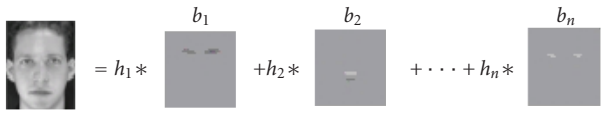


Figure 1: Face representation using space-localized basis images.

2. LOCAL FEATURE EXTRACTION TECHNIQUES

A number of recent algorithms aim at obtaining face representations using (a linear combination of) space-localized images roughly associated with the components of typical faces such as eyes, nose, and mouth, as in Figure 1.

The individual images form a (possibly nonorthogonal) basis, and the set of coefficients may be interpreted as the face "signature" related to the specific basis. In the following, we present the main characteristics of three distinct solutions for obtaining such localized images. The general setting is as follows: the available N training images are organized as a matrix $\mathbf{X}$, where a column consists of the raster-scanned p pixel values of a face. We denote by $\mathbf{B}$ the set of m basis vectors, and by $\mathbf{H}$ the matrix of projected coordinates of data matrix $\mathbf{X}$ onto basis $\mathbf{B}$. If the number of basis vectors is smaller than the length of the image vectors forming $\mathbf{X}$, we get dimensionality reduction. On the contrary, if the number of basis images exceeds training data dimensionality, we obtain overcomplete representations. As a consequence, we may write

$$\mathbf{X} \simeq \mathbf{BH}, \tag{1}$$

where $\mathbf{X} \in \mathbb{R}^{pxN}$, $\mathbf{B} \in \mathbb{R}^{pxm}$, and $\mathbf{H} \in \mathbb{R}^{mxN}$. Different linear techniques impose specific constraints on $\mathbf{B}$ and/or $\mathbf{H}$, and some yield spatially localized basis images.

2.1. Local nonnegative matrix factorization

Nonnegative matrix factorization (NMF) [15] has been recently introduced as a linear projection technique that imposes nonnegativity constraints on both $\mathbf{B}$ and $\mathbf{H}$ matrices during learning. The method resembles matrix decompositions techniques such as positive matrix factorization [16], and has found many practical applications including chemometric or remote-sensing data analysis. The basic idea is that only *additive* combinations of the basis vectors are allowed, following the intuitive scheme of combining parts to form a whole. Referring to (1), NMF imposes the following restrictions:

$$\mathbf{B},\mathbf{H} \geq \mathbf{0}. \tag{2}$$

Unlike simulation results reported in [15], the images provided by NMF, when applied to human faces, still maintain a holistic aspect, particularly in case of poorly aligned images, as was previously noted by several authors. In order to improve localization, a local version of the algorithm has been proposed in [12] that imposes the following additional constraints: (a) maximum sparsity of coefficients matrix $\mathbf{H}$; (b) maximum expressiveness of basis vectors $\mathbf{B}$ (keep only those coefficients bearing the most important information); (c) maximum orthogonality of $\mathbf{B}$. The following equations describe the updating procedure for $\mathbf{B}$ and $\mathbf{H}$:

$$\begin{aligned} H_{aj} &\longleftarrow \sqrt{H_{aj} \sum_i [\mathbf{B}^T]_{ai} \frac{X_{ij}}{[\mathbf{BH}]_{ij}}}, \\ B_{ia} &\longleftarrow B_{ia} \sum_j \frac{X_{ij}}{[\mathbf{BH}]_{ij}} [\mathbf{H}^T]_{ja}, \\ B_{ia} &\longleftarrow \frac{B_{ia}}{\sum_j B_{ja}}. \end{aligned} \tag{3}$$

Examples of basis vectors obtained by performing LNMF on AR database images are presented in Figure 2(a).

2.2. Independent components analysis

Natural images are highly redundant. A number of authors argued that such redundancy provides knowledge [17], and that the role of the sensory system is to develop factorial representations in which the dependencies between pixels are separated into statistically independent components. While in PCA and LDA the basis vectors depend only on pairwise relationships among pixels, it is argued that higher-order statistics are necessary for face recognition, and ICA is an example of a method sensible to such statistics. Basically, given a set of linear mixtures of several statistically independent components, ICA aims at estimating both the mixing matrix and the source components based on the assumption of statistical independence.

There are two distinct possibilities to apply ICA for face recognition [13]. The one of interest from the perspective of the present paper organizes the database into a large matrix, whereas *every image is a different column*. In this case, images are random variables and pixels are outcomes (independent trials). We look for the independence of images or functions of images. Two i and j images are independent if, when moving across pixels, it is not possible to predict the value taken by a pixel on image i based on the value taken by the same pixel on image j. The specific computational procedure includes two steps [13].

(a) Perform PCA to project original data into a lower-dimensional subspace: this step both eliminates less significant information and simplifies further processing, since resulting data is decorrelated (and only higher-order dependencies are to be separated by ICA). Let $\mathbf{V}_{\mathrm{PCA}} \in \mathbb{R}^{pxm}$ be the matrix whose columns represent the first m eigenvectors of the set of N training images, and $\mathbf{C} \in \mathbb{R}^{mxN}$ the corresponding PCA coefficients matrix, we may write $\mathbf{X} = \mathbf{V}_{\mathrm{PCA}} * \mathbf{C}$.

(b) ICA is actually performed on matrix $\mathbf{V}_{\mathrm{PCA}}^T$, and the independent basis images are computed as $\mathbf{B} = \mathbf{W} * \mathbf{V}_{\mathrm{PCA}}^T$, where the *separating matrix* $\mathbf{W}$ is obtained with the InfoMax method [18] (since directly maximizing the independence condition is difficult, the general

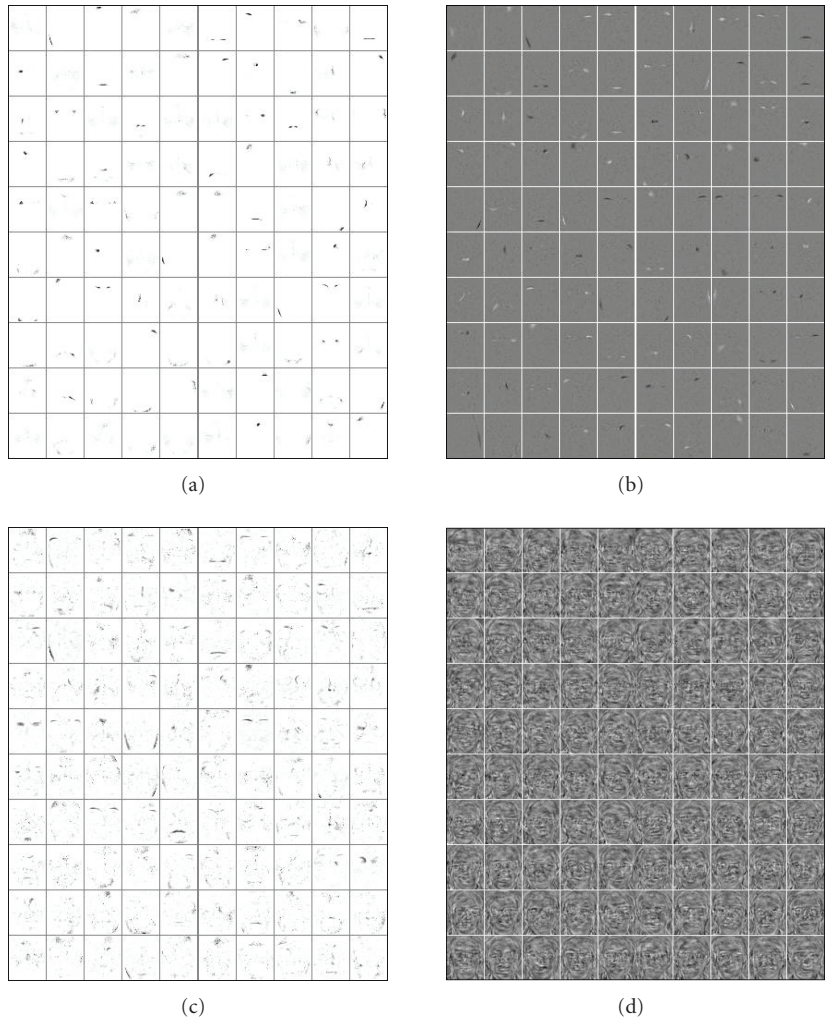


Figure 2: Examples of basis vectors for AR image database: (a) LNMF; (b) ICA; (c) NMFsc; (d) LPP.

approach of most ICA methods aims at optimizing an appropriate objective function whose extreme occurs when the unmixed components are independent; several distinct types of objective functions are commonly used, e.g., InfoMax algorithm maximizes the entropy of the components). The set of projected coordinates on ICA subspace (the set of coefficients that linearly combine the basis images in order to reconstruct the original face images) is computed as $\mathbf{H}^T = \mathbf{C} * \mathbf{W}^{-1}$.

Due to somehow contradictory comparative results between ICA and PCA presented in the literature, a systematic analysis has been reported in [19] in terms of algorithms and architectures used to implement ICA, the number of subspace dimensions, distance metric, and recognition task (facial identity versus expression). Results indicate that specific ICA design strategies are superior to standard PCA, although the task to be performed remains the most important factor. Examples of basis images obtained by ICA-InfoMax approach are presented in Figure 2(b) (Matlab code is available at http://inc.ucsd.edu/~marni/code.html).

2.3. NMF with sparseness constraints

A random variable is called *sparse* if its probability density is highly peaked at zero and has heavy tails. Within the general setting expressed by (1), sparsity is an attribute of the activation vectors grouped in the lines of coefficients matrix **H**, the set of basis images arranged in the columns of **B**, or both. While standard NMF does yield a sparse representation of the data, there is no effective way to control the degree of sparseness. Augmenting standard NMF with the sparsity concept proved useful for dealing with overcomplete representations (i.e., cases where the dimensionality of the space spanned by decomposition is larger than the effective dimensionality of the input space). While not present in standard NMF definition, sparsity is taken into account in LNMF and nonnegative sparse coding [14]. In fact, the latter enables the control over the (relative) sparsity level in **B** and **H** by defining an objective function that combines the goals of minimizing the reconstruction error and maximizing the sparseness level. Unfortunately, the optimal values of the parameters describing the algorithm are set by extensive

trial-and-error experiments. This shortcoming is eliminated in a more recent contribution of the same author, which proposed a method termed *NMF with sparseness constraints (NMFsc)* [14]. Sparseness of an n-dimensional vector $\mathbf{x}$ is defined as follows:

$$\text{sparseness}\,(\mathbf{x}) = \frac{\sqrt{n} - \left(\sum |x_i|\right) \Big/ \sqrt{\sum x_i^2}}{\sqrt{n} - 1}. \tag{4}$$

The algorithm proceeds by iteratively performing a gradient descent step on the (Euclidean distance type) objective function, as in (5), followed by projecting the resulting vectors onto the constraint space:

$$\mathbf{B} = \mathbf{B} - \mu_{\mathrm{B}}(\mathbf{WH} - \mathbf{X})\mathbf{H}^T. \tag{5}$$

The projection operator is the key element of the whole processing procedure, which sets explicitly the L_1 and L_2 norms of the basis components, and is fully described in [14]. Examples of basis images obtained after applying NMFsc on AR face database images are presented in Figure 2(c) (Matlab code is available at http://www.cs.helsinki.fi/patrik.hoyer/).

2.4. Locality-preserving projections

Linear subspace projection techniques such as PCA or LDA are unable to approximate accurately data lying on nonlinear submanifolds hidden in the face space. Although several nonlinear solutions to unveil the structure of such manifolds have been proposed (Isomap [20], LLE [21], Laplacian eigenmaps [22]), these are defined only on the training set data points, and the possibility of extending them to cover new data remains largely unsolved (efforts towards tackling this issue are reported in [23]). An alternative solution is to use methods aiming at preserving the *local structure* of the manifold after subspace projection, which should be preferred when nearest neighbor classification is to be subsequently performed. One such method is Locality-preserving projections (LPPs) [24]. LPP represents a linear approximation of the nonlinear Laplacian eigenmaps introduced in [22]. It aims at preserving the intrinsic geometry of the data by forcing neighboring points in the original data space to be mapped into closely projected data. The algorithm starts by defining a similarity matrix $\mathbf{S}$, based on a (weighted) k nearest neighbors graph, whose entry $\mathbf{S}_{ij}$ represents the edge between training images (graph nodes) $\mathbf{x}_i$ and $\mathbf{x}_j$. Gaussian-type weights of the form $\mathbf{S}_{ij} = e^{-(\|\mathbf{x}_i - \mathbf{x}_j\|^2)/\sigma}$ have been proposed in [24], although other choices (e.g., cosine type) are also possible. Based on matrix $\mathbf{S}$, a special objective function is constructed, enforcing the locality of the projected data points by penalizing those points that are mapped far apart. Basically, the approach reduces to finding a minimum eigenvalue solution to the following generalized eigenvalue problem:

$$\mathbf{XLX}^T\mathbf{b} = \lambda\mathbf{XDX}^T\mathbf{b}, \tag{6}$$

where $\mathbf{D} = \sum_i \mathbf{S}_{ij}$ and $\mathbf{L} = \mathbf{D} - \mathbf{S}$ (Laplacian matrix). The components of the subspace projection matrix $\mathbf{B}$ are the eigenvectors corresponding to the smallest eigenvalues of the problem above.

Rigorous theoretical grounds are related tooptimal linear approximations to the eigenfunctions of the Laplace-Bertrami operator on the manifold and are extensively presented in [24] (Matlab code is available at http://people.cs.uchicago.edu/~xiaofei). When applied to face image analysis, the method yields the so-called Laplacian faces, examples of which are presented in Figure 2(d).

Remark 1. Another interesting manifold learning algorithm called OPRA (orthogonal projection reduction by affinity) has been recently proposed [25], which also starts by constructing a weighted graph that models the data space topology. This affinity graph is built in a manner similar to the one used in local linear embedding (LLE) technique [21], and expresses each data point as a linear combination of (a limited number of) neighbors. The advantage of OPRA over LLE is that the mapping between the original data and the projected one is made explicit through a linear transformation, whereas in LLE this mapping is implicit, making it difficult to generalize to new test data. Compared to LPP, OPRA preserves not only the locality but also the geometry of local neighborhoods. Moreover, the basis vectors obtained by performing OPRA are orthogonal, whereas projection directions obtained by LPP are not. When class labels are available, as in our case, the algorithm is to be used in its supervised version, namely an edge is present between two nodes in the affinity graph only if the two corresponding data samples belong to the same class.

3. EXPERIMENTAL RESULTS

3.1. Image database preprocessing

AR database contains images of 116 individuals (63 males and 53 females). Original images are 768×576 pixels in size with 24-bit color resolution. The subjects were recorded twice at a 2-week interval, and during each session, 13 conditions with varying facial expressions, illumination, and occlusion were used. In Figure 3, we present examples from this database. As in [26], we used as training images two neutral poses of each person captured on different days (labeled $\mathrm{AR01}_1$ and $\mathrm{AR01}_2$ in Figure 3),while the testing set consists of pairs of images for the remaining 12 conditions, AR02, ..., AR13, respectively. More specifically, images AR02, AR03, and AR04 are used for testing the performances of the analyzed techniques to deal with expression variation (smile, anger, and scream), images AR05, AR06, and AR07 are used for illumination variability, and the rest of the images are related to occlusion (eyeglasses and scarf), with variable illumination conditions. The subset of the AR database is the same as in [26], and was kindly provided by the author. First, pose normalization has been applied in order to align all database faces, according to the (manually) localized eye positions. Next, only part of a face inside an elliptical region was selected, in order to avoid the influence of the background. The size of each reduced image is 40×48 pixels, and when considering the elliptical region only, each image

FIGURE 3: Example of one individual from the AR face database: (1) neutral, (2) smile, (3) anger, (4) scream, (5) left light on, (6) right light on, (7) both lights on, (8) sunglasses, (9, 10) sunglasses left/right light, (11) scarf, (12, 13) scarf left/right light.

is represented using 1505 pixels. No illumination normalization procedure has been applied, since we are directly interested in a comparative analysis of the algorithms *per se* dealing with illumination variability (although preliminary tests using histogram equalized images indicate that recognition accuracy deteriorates in most cases).

Olivetti database comprises 10 distinct images of 40 persons, represented by 112×92 pixels, with 256 gray levels. All the images were taken against a dark homogeneous background with the subjects in an upright frontal position, with tolerance for some tilting and rotation of up to about 20 degrees. In order to enable comparisons with previously reported results, we randomly selected 5 images per person for the training set, the remaining 5 images were included in the test set, and average recognition rates over 20 distinct trials were computed.

3.2. Comparative performance analysis

In this section, we present simulation results for the algorithms described in Section 2. The performances are given in terms of recognition accuracy and are compared to results obtained by performing standard PCA. The design items taken into account are (a) the distance metric used: Euclidean (L2), Manhattan (L1), cos (cosine of the angle between the compared vectors, $\cos(\mathbf{x}, \mathbf{y}) = (\mathbf{x} \cdot \mathbf{y})/(\|\mathbf{x}\| \, \|\mathbf{y}\|)$); (b) projection subspace dimension: the dimension of the feature space, equal to the number of basis vectors used, is set to 50, 100, 150, and 200 dimensions.

In order to make the evaluation, we conducted a rank-based analysis as follows: for each image/dimension combination, we ordered the performance rank of each algorithm/distance measure combination (the highest recognition rate got rank 1, and so on) regardless of the subspace dimension. This yielded a total of 11 rank numbers for each case: expression variation, illumination variation, glasses, and scarf. Then, we computed a sum of ranks for each of the algorithms over all the cases, and ordered the results (the lowest sum indicates the best overall performance).

3.2.1. Facial expression recognition

The capacity of the methods to deal with expression variability was tested using images labeled AR02, AR03, and AR04, and results are presented in Table 1. Algorithm NMFsc using L_1 distance deals best with smile expression, while LNMF + L_1 and ICA + COS combinations give best results for smile and anger expressions, respectively. Recognition accuracies of up to 96% are obtained for AR02 and AR03 images, while 62.4% is reached for the most difficult task AR04. Rank analysis conducted on combined AR02, AR03, and AR04 images reveals that the LNMF + L_1 approach outperforms the other competitors, followed by ICA + L_1/L_2 algorithm, as presented in Table 2. Generally, greater basis dimensionality tends to be favored. L_1 norm yields the best results, followed by L_2 and the cosine metric. While performing second best for smile expression, standard PCA occupies a middle position on the combined expression rank analysis results.

3.2.2. Changing illumination conditions

Changing illumination conditions are reflected in images AR05, AR06, and AR07, and recognition performances are given in Table 3. The ICA-InfoMax approach ranks best on both individual tests and combined analysis, with accuracies of up to 98%, 97%, and 89%, respectively. Laplacian faces perform second best, followed by PCA. Greater basis dimensionality yields better results, while no distance metric is favored. Standard PCA is placed again on a middle position, better than LNMF and NMFsc algorithms. It is worth noting

Table 1: Recognition rates for AR database/expression variability.

	Expression											
	AR02				AR03				AR04			
	$m = 50$	$m = 100$	$m = 150$	$m = 200$	$m = 50$	$m = 100$	$m = 150$	$m = 200$	$m = 50$	$m = 100$	$m = 150$	$m = 200$
L2	83.7	72.2	64.5	82.4	89.7	92.3	93.1	95.3	41.4	46.1	39.7	49.5
L1	92.7	92.3	86.3	95.7	93.1	94	94.8	96.5	53	56.8	54.7	61.5
LNMF + cos	76	64.1	59.4	73.9	84.6	90.6	92.7	93.6	34.2	38.9	33.3	41.8
L2	91	92.3	92.7	91.8	91.4	92.3	93.1	93.6	49.1	52.1	53	55.1
ICA + L1	91	92.7	91.8	91.4	93.1	93.6	93.6	94.4	51.7	55.1	55.5	57.2
cos	89.7	90.6	91.4	89.7	89.3	90.6	91	90.1	58.1	62.4	60.6	61.1
L2	79	91	92.7	93.1	67.9	85.9	89.7	88.4	29.5	38.9	38.9	44.4
NMFsc + L1	88.9	95.7	96.1	93.6	86.7	91.8	92.7	90.6	41.8	44	46.5	46.5
cos	73.5	88	91.8	91.8	65.8	85.4	91	89.3	26.9	37.1	38.9	45.7
Laplacian faces	73.9	87.2	89.7	89.7	83.7	91.4	91.8	91.4	17	30.8	29.5	30.8
PCA	91	94.4	95.3	95.7	88	89.7	89.7	90.6	47.4	52.5	52.5	52.5

Table 2: Rank-based analysis results.

Algorithm/distance	Expression rank	Illumination rank	Glasses rank	Scarf rank	Sum of ranks
ICA-cos	14	6	3	6	29
ICA-L1	10	8	9	3	30
ICA-L2	12	7	6	9	34
Laplacian	27	9	23	15	74
LNMF-L1	5	39	18	24	86
NMFsc-L1	13	31	22	31	95
PCA	15	25	24	39	103
NMFsc-cos	20	26	28	31	105
NMFsc-L2	22	29	30	27	108
LNMF-L2	17	40	37	32	126
LNMF-cos	23	34	38	32	127

that recognition accuracies are significantly different for left and right illumination directions, although the use of an appropriate illumination normalization procedure could have changed this conclusion.

3.2.3. Occlusion

Occlusion is one of the situations that hopefully should be better tackled by local-based techniques compared to holistic ones such as PCA. AR database provides two kinds of partially occluded images, using sunglasses (images AR08) and scarf (images AR11). Due to length constraints, we only present in Table 4 results for eyeglasses occlusion, although both cases show a significant general decrease of the recognition performances, especially when the illumination conditions are changing. Recognition accuracies do not exceed 47%, while differences between left and right illumination directions are maintained.

3.2.4. Pose variation

In Table 5 we give simulation results for the Olivetti database, which present significant pose variation, while illumination conditions are better controlled. LNMF + L_1 and OPRA faces method yield the best results, followed by PCA and ICA + COS, and all algorithms show rather limited dependence on the subspace dimension. A key observation related to using OPRA in its supervised version must be made: since the method relies on the assumption that each data point may be approximated by a linear combination of its k nearest neighbors belonging to the same class, we could not use this method in case of AR database, where only 2 training samples per class are available.

4. CONCLUSIONS

We conducted an extensive set of experiments in order to provide a comparative analysis of the recognition performances of several modern subspace projection algorithms in terms of distance metric used, number of selected features, and sources of variability on AR and Olivetti face databases. The study revealed that ICA implemented by the InfoMax algorithm seems best suited for face oriented tasks, outperforming clearly all other solutions in case of AR database. While explaining the exact reason for this remarkable performance needs further study, we may note that searching

Table 3: Recognition rates for AR database/illumination variability.

	Illumination											
	AR05				AR06				AR07			
	$m = 50$	$m = 100$	$m = 150$	$m = 200$	$m = 50$	$m = 100$	$m = 150$	$m = 200$	$m = 50$	$m = 100$	$m = 150$	$m = 200$
L2	17	29.5	36.3	25.6	11.9	13.6	11.9	8.9	2.1	6.4	3.4	1.7
L1	20	32.9	38.4	28.2	8.1	20	13.6	11.5	1.2	1.7	2.1	2.1
LNMF + cos	46.5	48.3	53.8	57.2	30.7	23	20.9	10.2	17	15.3	14.5	17
L2	95.3	97.4	97.4	98.3	89.3	92.7	93.6	93.1	73.9	79.5	80.3	79.9
ICA + L1	95.3	97	97.8	97.4	90.6	92.3	94.8	92.7	75.6	79	79.5	79
cos	95.7	97.4	97.4	97	94	97.4	97.8	97.4	88.4	89.3	89.3	88.9
L2	44	56	76	71.3	9.8	22.6	22.2	34.2	11.1	15.3	23.5	27.3
NMFsc + L1	43.1	53	73	76	9.4	26.5	17.9	32	5.1	10.6	19.6	20.9
cos	55.1	61.9	77.3	73.9	11.9	27.7	25.6	36.7	22.6	24.3	34.6	37.6
Laplacian faces	79.5	91.5	94.4	95.3	72.6	93.2	93.1	92.7	56.8	87.2	91.4	89.7
PCA	73.5	77.3	80.7	81.2	16.2	20.9	21.3	21.3	58.9	67	70	71.3

Table 4: Recognition rates for AR database/occlusion (sunglasses).

	Occlusion sunglasses											
	AR08				AR09				AR10			
	$m = 50$	$m = 100$	$m = 150$	$m = 200$	$m = 50$	$m = 100$	$m = 150$	$m = 200$	$m = 50$	$m = 100$	$m = 150$	$m = 200$
L2	7.7	6.8	5.5	7.2	5.1	5.5	3.4	3.8	5.1	2.5	3.8	3.8
L1	22.2	20.5	17	28.6	10.6	14.1	12.8	14.1	9	6.4	5.1	7.7
LNMF + cos	8.1	6	2.5	6.8	4.2	4.7	3	3.8	4.7	2.1	2.1	3
L2	28.2	34.6	34.6	35.9	26	27.7	29	29.5	26	29	29.5	30.3
ICA + L1	26.9	29.5	30.7	32.9	27.3	25.2	26	28.6	26.5	25.6	27.3	27.3
cos	39.3	43.6	45.3	47.4	36.3	38.9	40.6	40.6	31.6	36.7	36.7	38
L2	10.2	9.8	11.5	17	5.5	8.1	7.7	11.5	6.4	6.8	8.1	8.5
NMFsc + L1	18.3	14.5	15.3	23.9	9.4	9.4	8.1	11.1	5.5	6.4	7.7	9.4
cos	9.8	9.4	9.8	18.3	4.2	7.2	7.2	11.1	5.1	6	6.4	7.7
Laplacian faces	8.9	15	17.9	18.3	4.7	6.8	11.5	12.4	4.7	8.9	8.9	9.4
PCA	8.5	8.5	10.2	11.1	11.5	12.4	13.2	13.2	8.9	9.8	9.4	9.8

Table 5: Recognition rates for Olivetti database.

	$m = 50$	$m = 100$	$m = 150$	$m = 200$
L2	90.4	93.4	93.2	92.8
L1	92.3	95.1	94.4	94.3
LNMF + cos	89.1	92.9	91.7	91.1
L2	92	92.7	92.4	93
ICA + L1	92.3	93.3	92.8	93.7
cos	93.4	94.3	93.2	93.7
L2	89	91	89.9	90
NMFsc + L1	92	90.5	91.6	90.5
cos	91	92	90.8	92
Laplacian faces	91.1	90.7	89.9	90.7
OPRA faces	94.2	94.9	95	92.8
PCA	93.9	94.4	93.3	94.3

for most *informative features* (instead for most expressive ones, as in PCA, or most discriminant, as in LDA) has been previously proposed in the literature. Moreover, considering recognition performances reported in an independent study [26], we may conclude that ICA-InfoMax compares favorably with two leading computer vision techniques, namely Local Feature Analysis [27], and Bayesian PCA [28], where a similar experimental setup based on AR database was used.

Based on overall results it is worth noting that, except for expression recognition, manifold learning algorithms rank amongst the top performers. Moreover, PCA also compares favorably to most local representations (except for the occlusion tasks), confirming the conclusions from [29].

Some other conclusions agree with previously reported results, namely cosine and L_1 metrics are almost always superior to L_2, and the dependence of the recognition rates on the projection subspace dimension is not always clear (although larger dimensions tend to be generally favored).

Some important aspects must be tackled if these approaches are to become important tools in face oriented applications. Reliable selection of significant basis vectors is still an open problem, if the number of training images per class is small. Basis vectors exhibiting invariance to common transformations such as translations and in-plane rotations

are desirable. Finally, a key problem to be further addressed is the identification of the conditions under which correct decompositions of faces into significant/generic parts emerge [30].

REFERENCES

[1] S. G. Kong, J. Heo, B. R. Abidi, J. Paik, and M. A. Abidi, "Recent advances in visual and infrared face recognition—a review," *Computer Vision and Image Understanding*, vol. 97, no. 1, pp. 103–135, 2005.

[2] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recognition: a literature survey," *ACM Computing Surveys*, vol. 35, no. 4, pp. 399–458, 2003.

[3] J. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, "Face recognition using kernel direct discriminant analysis algorithms," *IEEE Transactions on Neural Networks*, vol. 14, no. 1, pp. 117–126, 2003.

[4] M.-H. Yang, "Kernel eigenfaces vs. kernel fisherfaces: face recognition using kernel methods," in *Proceedings of the 5th IEEE International Conference on Automatic Face and Gesture Recognition (FGR '02)*, pp. 215–220, Washington, DC, USA, May 2002.

[5] J. Yang, A. F. Frangi, J.-Y. Yang, D. Zhang, and Z. Jin, "KPCA plus LDA: a complete kernel fisher discriminant framework for feature extraction and recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 2, pp. 230–244, 2005.

[6] J. Yang, X. Gao, D. Zhang, and J.-Y. Yang, "Kernel ICA: an alternative formulation and its application to face recognition," *Pattern Recognition*, vol. 38, no. 10, pp. 1784–1787, 2005.

[7] B. Heisele, P. Ho, J. Wu, and T. Poggio, "Face recognition: component-based versus global approaches," *Computer Vision and Image Understanding*, vol. 91, no. 1-2, pp. 6–21, 2003.

[8] S. Lucey and T. Chen, "A GMM parts based face representation for improved verification through relevance adaptation," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '04)*, vol. 2, pp. 855–861, Washington, DC, USA, June-July 2004.

[9] X. He, S. Yan, Y. Hu, P. Niyogi, and H.-J. Zhang, "Face recognition using Laplacianfaces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 3, pp. 328–340, 2005.

[10] J. Zhang, S. Z. Li, and J. Wang, "Manifold learning and applications in recognition," in *Intelligent Multimedia Processing with Soft Computing*, Springer, Heidelberg, Germany, 2004.

[11] FRVT 2002, 2004: Evaluation Report, http://www.frvt.org.

[12] S. Z. Li, X. W. Hou, H. J. Zhang, and Q. S. Cheng, "Learning spatially localized, parts-based representation," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '01)*, vol. 1, pp. 207–212, Kauai, Hawaii, USA, December 2001.

[13] M. S. Bartlett, J. R. Movellan, and T. J. Sejnowski, "Face recognition by independent component analysis," *IEEE Transactions on Neural Networks*, vol. 13, no. 6, pp. 1450–1464, 2002.

[14] P. O. Hoyer, "Non-negative matrix factorization with sparseness constraints," *Journal of Machine Learning Research*, vol. 5, pp. 1457–1469, 2004.

[15] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, 1999.

[16] P. Paatero and U. Tapper, "Positive matrix factorization: a non-negative factor model with optimal utilization of error estimates of data values," *Environmetrics*, vol. 5, no. 2, pp. 111–126, 1994.

[17] H. B. Barlow, "Unsupervised learning," *Neural Computation*, vol. 1, no. 3, pp. 295–311, 1989.

[18] A. J. Bell and T. J. Sejnowski, "An information-maximization approach to blind separation and blind deconvolution," *Neural Computation*, vol. 7, no. 6, pp. 1129–1159, 1995.

[19] B. A. Draper, K. Baek, M. S. Bartlett, and J. R. Beveridge, "Recognizing faces with PCA and ICA," *Computer Vision and Image Understanding*, vol. 91, no. 1-2, pp. 115–137, 2003.

[20] J. B. Tenenbaum, V. de Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[21] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[22] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural Computation*, vol. 15, no. 6, pp. 1373–1396, 2003.

[23] Y. Bengio, J.-F. Paiement, P. Vincent, O. Delalleau, N. Le Roux, and M. Ouimet, "Out-of-sample extensions for LLE, isomap, MDS, eigenmaps, and spectral clustering," in *Proceedings of the Annual Conference on Neural Information Processing Systems 16 (NIPS '03)*, pp. 177–184, Vancouver, Canada, December 2003.

[24] X. He and P. Niyogi, "Locality preserving projections," in *Proceedings of the Annual Conference on Neural Information Processing Systems 16 (NIPS '03)*, Vancouver, Canada, December 2003.

[25] E. Kokiopoulou and Y. Saad, "Face recognition using OPRA -faces," in *Proceedings of the 4th International Conference on Machine Learning and Applications (ICMLA '05)*, vol. 2005, pp. 69–74, Los Angeles, Calif, USA, December 2005.

[26] D. Guillamet and J. Vitrià, "Classifying faces with non-negative matrix factorization," in *Proceedings of the 5th Catalan Conference on Artificial Intelligence (CCIA '02)*, vol. 2504, pp. 24–31, Castelló de la Plana, Spain, 2002.

[27] P. S. Penev and J. J. Atick, "Local feature analysis: a general statistical theory for object representation," *Network: Computation in Neural Systems*, vol. 7, no. 3, pp. 477–500, 1996.

[28] B. Moghaddam and A. Pentland, "Probabilistic visual learning for object representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 696–710, 1997.

[29] K. W. Bowyer and P. J. Phillips, *Empirical Evaluation Techniques in Computer Vision*, Wiley-IEEE Computer Society Press, Hoboken, NJ, USA, 1998.

[30] D. Donoho and V. Stodden, "When does non-negative matrix factorization give a correct decomposition into parts?" in *Proceedings of the Annual Conference onNeural Information Processing Systems 16 (NIPS '03)*, Vancouver, Canada, December 2003.

Hindawi Publishing Corporation
EURASIP Journal on Image and Video Processing
Volume 2007, Article ID 25409, 8 pages
doi:10.1155/2007/25409

Research Article

View Influence Analysis and Optimization for Multiview Face Recognition

Won-Sook Lee[1] and Kyung-Ah Sohn[2]

[1] *School of Information Technology and Engineering, University of Ottawa, Ottawa, Canada K1N6N5*
[2] *Computer Science Department, Carnegie Mellon University, Pittsburgh, PA 15213-3891, USA*

Received 1 May 2006; Revised 20 December 2006; Accepted 24 June 2007

Recommended by Christophe Garcia

We present a novel method to recognize a multiview face (i.e., to recognize a face under different views) through optimization of multiple single-view face recognitions. Many current face descriptors show quite satisfactory results to recognize identity of people with given limited view (especially for the frontal view), but the full view of the human head has not yet been recognizable with commercially acceptable accuracy. As there are various single-view recognition techniques already developed for very high success rate, for instance, MPEG-7 advanced face recognizer, we propose a new paradigm to facilitate multiview face recognition, not through a multiview face recognizer, but through multiple single-view recognizers. To retrieve faces in any view from a registered descriptor, we need to give corresponding view information to the descriptor. As the descriptor needs to provide any requested view in 3D space, we refer to it as "3D" information that it needs to contain. Our analysis in various angled views checks the extent of each view influence and it provides a way to recognize a face through optimized integration of single view descriptors covering the view plane of horizontal rotation from -90° to 90° and vertical rotation from -30° to 30°. The resulting face descriptor based on multiple representative views, which is of compact size, shows reasonable face recognition performance on any view. Hence, our face descriptor contains quite enough 3D information of a person's face to help for recognition and eventually for search, retrieval, and browsing of photographs, videos, and 3D-facial model databases.

1. INTRODUCTION

Face recognition techniques have started to be used as commercial products in the last few years, especially on the frontal images, but with certain constraints such as indoor environment, controlled illumination, and small degree of facial expression as can be seen in many literatures, for example, in a classic survey paper by Samal and Iyengar [1]. Face recognition is composed of two main steps, registration and retrieval. We register a person's face in a certain form, and we retrieve the person's face out of many people's faces. One problem we want to raise in this paper is what is the optimized way to determine how many views and which angle we need to register the person to retrieve the person in any angle. As an effort to make more practical systems, various researches have been performed to detect and recognize faces in arbitrary poses or views. However, those approaches using statistical learning methods [2–4] reveal limitation to satisfy practically acceptable recognition performance. Novel view generation using 3D morphable model approach [5] shows quite reasonable success rate in many different views, but it still depends on the database of 3D generic models to build the linear interpolation of a given person and also it needs high computational costs with very complicated algorithms behind. Recently, 3D face model from direct 3D scanning could be used for face recognition [6–8], but the successful reconstruction is not always guaranteed in real time and the recognition rate is not yet as good as the 2D-image-based face recognition. In addition, the acquisition of the data is not always as easy as images and we still need more robust and stable sensing equipment to get meaningful recognition applications. In short, multiview face recognition has still a lot lower recognition rate compared to single view recognition.

As a representative method of the currently available 2D-based face descriptor, MPEG-7 advanced face recognizer [9, 10] shows quite satisfactory results to recognize identity of people with given single view, and it especially shows good performance on the frontal view. However, the single-view-based face descriptor, as it allows only one view to build its

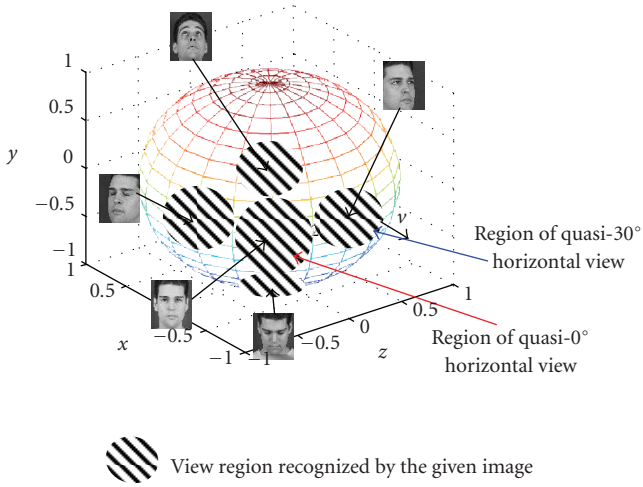


Figure 1: Single view recognition of the view-sphere surface.

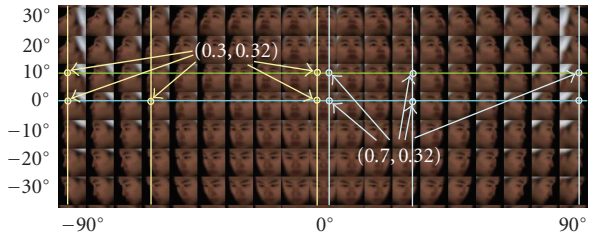
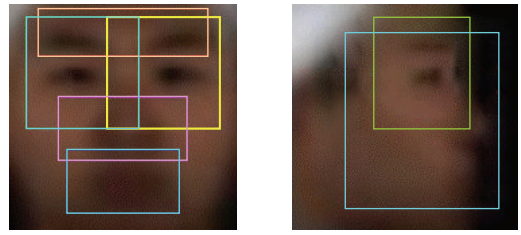


Figure 2: Eye positions on view mosaic of faces from 108 rendered images of 3D facial mesh models. Left eye position keeps (0.7, 0.32) for positive horizontal rotation while right eye position does (0.3, 0.32) for negative horizontal rotation when width and height of the image are considered as 1.0.

descriptor, causes problems to recognize other views. Nevertheless, it still allows nearby frontal views recognizable with desirable success rate.

In this paper, we present a novel face descriptor based on multiple single-view recognition, which aims to contain multiview 3D information of a person to help for face recognition in any view. In this scenario, we save or register the face descriptor as unique information of each person, and when we have a query face image in arbitrary view, we can identify the person's identity by comparing the registered descriptors with the one extracted from the query image. To retrieve such 3D information of a face to be recognizable in any view, we propose a method to extend the traditional 2D-image-based face recognition to 3D by combining multiple single views. We take a systematic extension to build 3D information using multiviews and perform optimization of the descriptor in respect of the number and the choice of views to be registered. In the following sections, we first describe the concept of multiview 3D face descriptor, and then show how to optimize multiple single views to build 3D information using our newly proposed "quasiview" concept, an extended term of quasifrontal, which measures the influence power of a certain view to nearby views. Experimental results then follow.



(a) Five subregion definitions on View (0°, 0°)

(b) Two subregion definitions on View (80°, 0°)

Figure 3: Subregion definition depending on views superimposed on the center face of our database.

2. MULTIVIEW 3D FACE DESCRIPTOR

The new descriptor we propose is called multiview 3D face descriptor, which is supposed to have sufficient 3D information of a face by describing the face as a mosaic of many one-views as shown in Figure 1. This multiview 3D face descriptor aims to cover any view between horizontal rotation from −90° to 90° and vertical rotation from −30° to 30°. We note the range of such horizontal and vertical views as $[-90^\circ \cdots 90^\circ]$ and $[-30^\circ \cdots 30^\circ]$, respectively. The notation of $[\cdot]$ is used to refer the range while $(\cdot)$ used for a position.

There are a few issues we encounter for the extension of the conventional single view descriptor to multiview version.

(i) *DB collection for training/test:* there are not yet enough data to be used for research on multiview face recognition. Most face database such as PIE, CMU, and YALE has been built mainly for frontal views even though nonfrontal face images are more usual.

(ii) *Multiview face detector:* to recognize a person from face images, we first need to detect faces on photographs, which is a rough alignment process.

(iii) *View estimator:* the view of the facial images should be estimated.

(iv) *Face alignment*: faces are then aligned in predefined location.

(v) *Feature extraction:* we extract features possibly depending on views.

(vi) *Descriptor optimization:* we intend to produce efficient descriptor containing views on horizontal rotation $[-90^\circ \cdots 90^\circ]$ and vertical rotation $[-30^\circ \cdots 30^\circ]$.

For DB generation, we could use 3D facial mesh models and render them to get face images in arbitrary views. For the experiment, the 3D facial mesh models of 108 subjects are used and their rendered images are used for training and test with 50/50 ratio. The database we use for the experiment is described in our previous work [11, 12] as well as pose estimation and feature detection. In this paper, we focus on the last two issues of feature extraction and descriptor optimization considering the various studies about multiview face detections and view estimations. The most naïve idea to create multiview descriptor from a single-view one is the

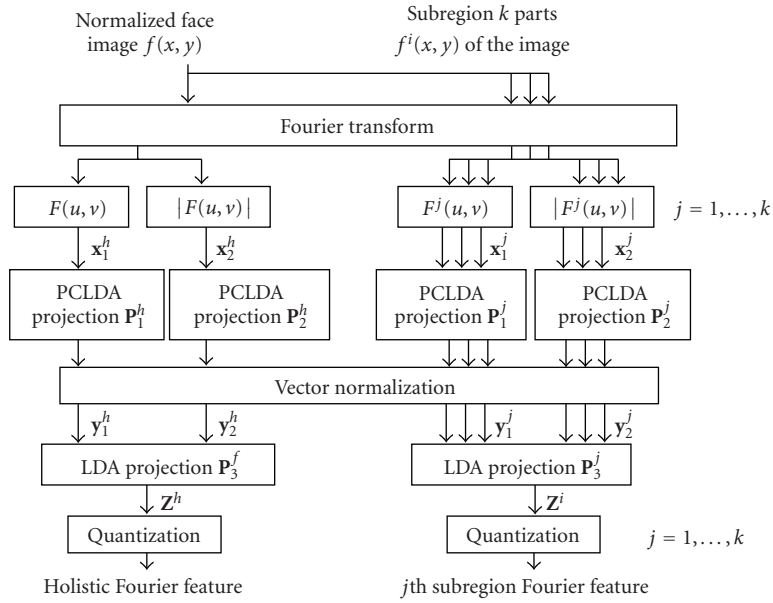


Figure 4: Feature extraction used for multiview 3D face descriptor.

simple integration of N uniformly distributed single view descriptors. If we register every 10° apart, that is, if we use every face image 10° apart for our descriptor, we have to register 19×7 views to cover the view space of horizontal rotation $[-90^\circ \cdots 90^\circ]$ and vertical rotation $[-30^\circ \cdots 30^\circ]$. Then this very naïve descriptor would result in size $133\times$ single view descriptor size, which becomes too big to be used in practice. Moreover, we could take advantage of the possibility that some view regions might have larger coverage than others so that we may need smaller number of views to describe those regions. While the descriptor optimization is one of the important steps for transition from single view to multiview face descriptor, there has not been until now any published result in this direction to the best of our knowledge. Here, we aim to make use of our learning from frontal-view face descriptors that a registered front view can be used to retrieve nearby frontal views (quasifrontal) with high success rate. Hence, we extend the concept of quasifrontal to quasiview and introduce some useful terms as follows.

(1) *View mosaic*. Mosaic of views 10° apart covering horizontal rotation $[-X^\circ \cdots X^\circ]$ and vertical rotation $[-Y^\circ \cdots Y^\circ]$. Here we choose $X = 90$ and $Y = 30$. It can be visualized as shown in Figure 2. This view mosaic is corresponding to any view (i.e., 3D) of a person wherever the face is at least half. It is used later on to check "quasiview" for each view in the view mosaic.

(2) *Quasiview with error rate K*. It is an extension of quasifrontal, from the frontal view to general views. For instance, quasiview V^q of a given (registered) view V with error rate K means that faces on view V^q can be retrieved using a registered face in view V with expected error rate less than or equal to K. This will be explored in Section 5

3. LOCALIZATION OF FACES IN MULTIVIEW

To use face images for training or as a query, we need to extract and normalize facial region. According to common practice, positions of two eyes are used for normalization such that the normalized image contains enough information of the face but excludes unnecessary background. The detailed localization specification is defined as follows.

(1) Size of images: 56×56.

(2) Positions of two eyes in the front view are on $(0.3, 0.32)$ and $(0.7, 0.32)$ when width and height are considered as 1.0. Here $(,)$ is used for (x, y) coordinates where the numbers are between 0 and 1.

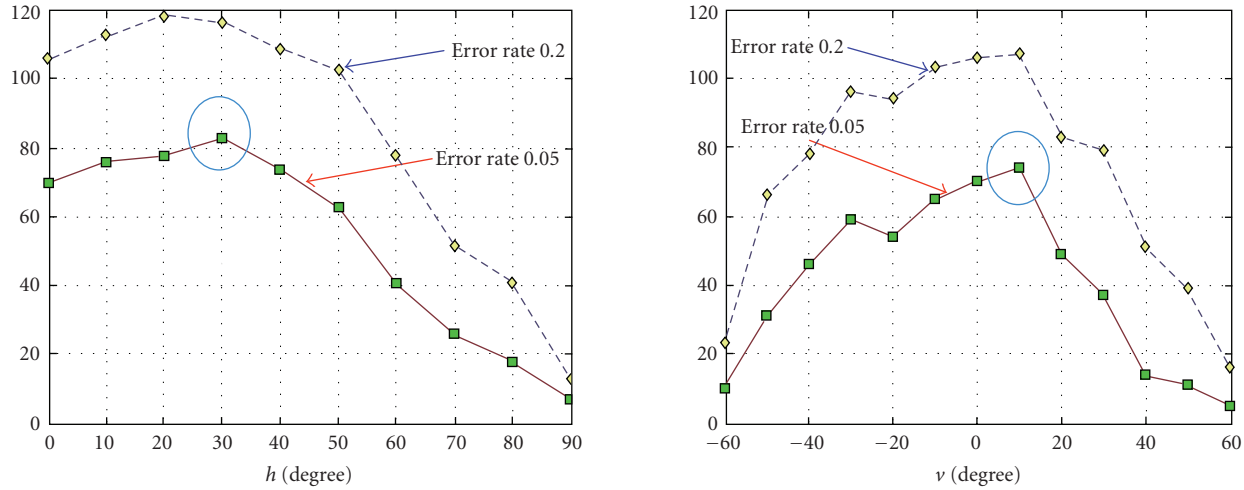
(3) Left eye position of the positive horizontal rotation keeps $(0.7, 0.32)$ while right eye position of the negative rotation does $(0.3, 0.32)$.

(4) Vertical rotation has the same eye positions as the ones on zero vertical rotation images.

Figure 2 summarizes the view mosaic of resulting localized images for our view space of horizontal rotation $[-90^\circ \cdots 90^\circ]$ and vertical rotation $[-30^\circ \cdots 30^\circ]$.

4. FEATURE EXTRACTION

As an example of single view face descriptor, we use the MPEG-7 advanced face recognition descriptor (AFR) [9] which showed best performance in retrieval accuracy, speed and data size as benchmarked by MPEG-7. More details can be found in MPEG document [9]. However, our focus in this paper is to show how to build optimized integration of multiple views to recognize a face in any view based on single-view face recognizers, so any single view face recognizer can be used instead of MPEG-7 AFR.



(a) Horizontal rotation (b) Vertical rotation

FIGURE 5: Quasiview sizes with horizontal and vertical rotations. The x-axis in (a) and (b) represents the degree of horizontal and vertical rotation, respectively, and the y-axis shows the number of neighboring views which could be recognized by registering the view in x-axis when allowed certain error rate (0.02 for blue plot, and 0.05 for red plot).

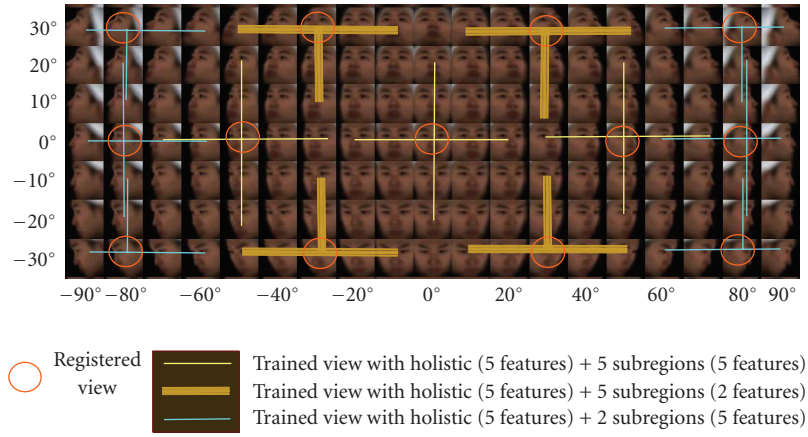


FIGURE 6: Views used for training and registration. 13 representative quasiviews are selected and used for training, and hence for registration. The number of used features (especially, the features for subregions) also varies depending on the view.

For our experiment, MPEG-7 AFR is modified to adapt to be multiview. AFR basically extracts features both in Fourier space and luminance space. In the Fourier space, features are extracted from the whole face, and luminance space extracts features from both the whole face and five subregions on the face as shown in Figure 3(a). We simplify, but also extend, this feature extraction algorithm to our *Subregion-based LDA on Fourier space* for multiview purpose. The biggest differences between the MPEG-7 AFR and our model are (i) feature extraction in luminance space is removed in our model; (ii) the subregion decomposition, which was in luminance space, is now in Fourier space and (iii) the number and positions of subregions are defined depending on a given view, for example, for new frontal views, we use the same five subregions as used in AFR, but for near profile view, we only use two subregions as shown in Figure 3(b). Figure 4 shows the overall feature extraction diagram. To summarize briefly, we first extract Fourier features from both the whole face image and each subregion of the image, and project all the features and their magnitudes using principle component—linear discriminant analysis (PCLDA) method. After normalizing the resulting vectors, we do additional LDA projection, and finally quantize them for descriptor efficiency. The first two modifications (i) and (ii) give more efficient feature extraction method with smaller descriptor size by extracting the same amount of information on a single space. The third modification (iii) is caused by the multiview extension. If we use the same definition of subregion used in the front view for the profile view, the background may seriously affect for recognition rate. So we define different subregion depending on views as shown in Figure 3.

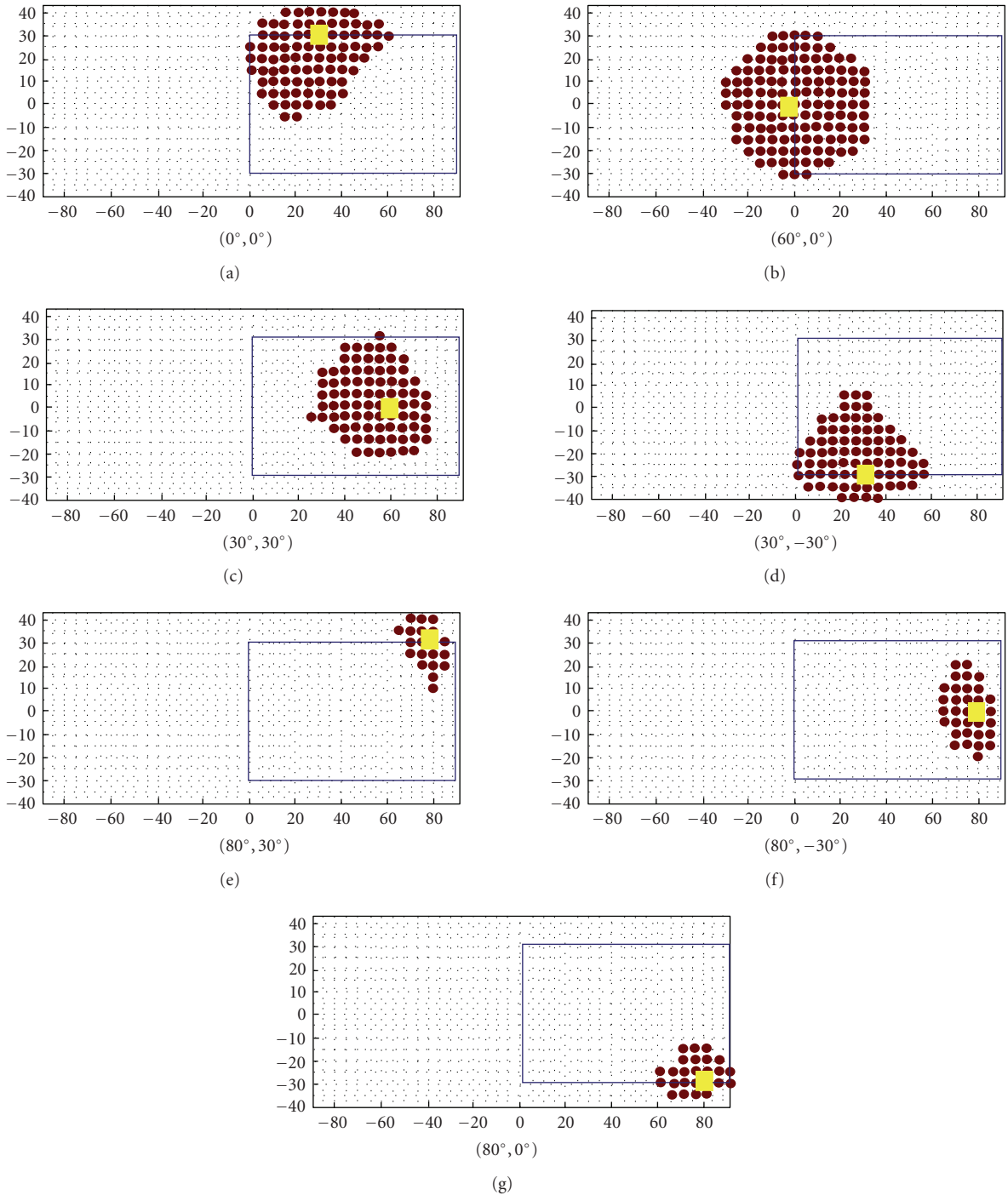


FIGURE 7: Representation of quasiviews. The x-axis and y-axis indicate the horizontal rotation from -90° to 90° and the vertical rotation from -40° to 40°, respectively. Big yellow spots represent the registered views and small red spots indicate corresponding quasiviews with error rate 0.05. The rectangles are the view region of interest in horizontal rotation $[0^\circ \cdots 90^\circ]$ and vertical rotation $[-30^\circ \cdots 30^\circ]$.

5. QUASIVIEW

Graham and Allinson [13] have calculated the distance between faces of different people over pose to predict the pose dependency of a recognition system. Using the average Euclidean distance between the people in the database over the pose angles sampled, they predicted that faces should be easiest to recognize around the 30° range and consequently, the best pose samples to use for an analysis should be concentrated around this range. Additionally, they expect that faces are easier to recognize at the frontal view $(0^\circ, 0^\circ)$ than the profile $(90^\circ, 0^\circ)$. Here, we use notation of (X°, Y°) to indicate a view with X° horizontal rotation and Y° vertical rotation.

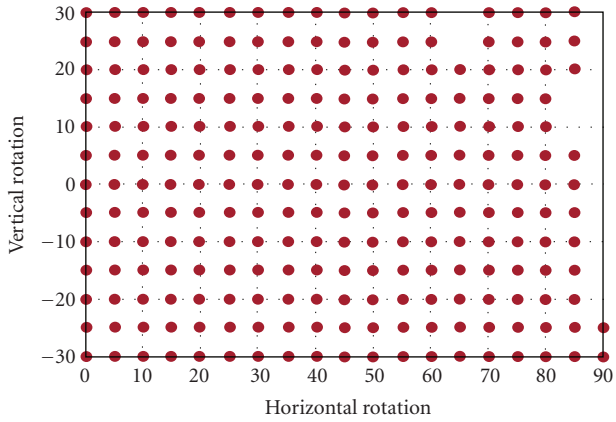


Figure 8: The region covered by 7 quasiviews in the view mosaic of horizontal rotation $[0^\circ \cdots 90^\circ]$ and vertical rotation $[-30^\circ \cdots 30^\circ]$ with error rate 0.05. Registration with 7 views covers 93.93% of the view space, which means that we can retrieve faces in any view represented in this plot from the registered descriptor within allowed error rate 0.05.

Note that they have checked only horizontal rotation of human heads.

We use the new concept of "quasiview" corresponding to the conventional "quasifrontal," which is a measurement of the influence of a registered view for recognition. To prove that quasiview size depends on the view, we performed experiments of quasiview inspection with accepted error rate 0.05, that is, we inspect the range of views that would be recognizable within error rate 0.05 given a view for registration. Figure 5 shows how the quasiview size varies with pure horizontal or vertical rotations of a head. To make fair comparison between different views, we extracted 24 holistic features (without using subregion features) for each view. And images of nearby views are also included in the training of certain view (i.e., in obtaining the PCLDA basis for each view). So for horizontal rotation, 9 views (the view of interest +8 nearby views) are used for training each view from $(0^\circ, 0^\circ)$ to $(70^\circ, 0^\circ)$, 8 training views for the view $(80^\circ, 0^\circ)$, and 7 training views for the view $(90^\circ, 0^\circ)$. For vertical rotation, 9 training views are used for each view from $(0^\circ, -40^\circ)$ to $(0^\circ, 40^\circ)$, 8 training views for the views $(0^\circ, -50^\circ)$ and $(0^\circ, 50^\circ)$, and 7 training views for the views $(0^\circ, -60^\circ)$ and $(0^\circ, 60^\circ)$. Figure 5 is obtained before adding neighboring images in training. Figure 6 can be helpful to understand which training views are used for each registered view while it reflects our result after optimization.

Figure 5 shows our quasiview measurements with synthetically created (rendered) images of 108 3D facial models by rotating them into various angles. We counted the number of nearby views which could be recognized when we registered a certain view using two kinds of accepted error rates 0.02 and 0.05. The result in Figure 5(a) shows very similar pattern with the graph showing the average distance between faces over view described in Graham and Allinson's paper [13]. The views $(20^\circ, 0^\circ) \sim (30^\circ, 0^\circ)$ have both the biggest quasiview size and the biggest Euclidean distance between the people in eigenspace among views

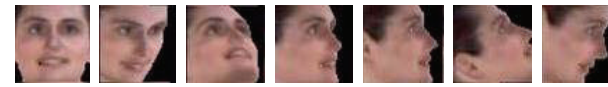


Figure 9: An example of registration. The views are needed in the registration step to recognize a face in the 93.93% of the view space where horizontal rotation $[0^\circ \cdots 90^\circ]$, vertical rotation $[-30^\circ \cdots 30^\circ]$, and their combined rotation of a head are allowed. It means that we can retrieve a face in various poses within allowed error rate 0.05 when we register only 7 views in a condition that a given face is symmetric.

$(0^\circ, 0^\circ), (10^\circ, 0^\circ), \ldots$, and $(90^\circ, 0^\circ)$. Figure 5(b) shows that the views $(0^\circ, 0^\circ) \sim (0^\circ, 10^\circ)$ have the biggest quasiview size among views $(0^\circ, -60^\circ), (0^\circ, -60^\circ), \ldots$, and $(0^\circ, 60^\circ)$. The views of heading downward have bigger quasiview size than ones of heading upward and it makes us guess that it might be easier to recognize people when they look downward more than they look upward.

6. DESCRIPTOR OPTIMIZATION

Based on our study to check the quasiview size on horizontal and vertical rotated heads, we now optimize the multiview 3D face descriptor by choosing several representative views and recording the corresponding view specific features together. We have used the following selection criteria for registration views: we register views (i) with bigger quasiview size for cost effect; (ii) which appear a lot in practice through target environment analysis, for example, ATM, door access control; (iii) considering efficient integration of quasiviews covering the big region in view-mosaic; (iv) and which are easy to register or easy to obtain. This choice is empirical and we focus on covering the bigger range of face views with more efficient face view registration. Remembering that our features are extracted from PCLDA projections, we can select the dimension of resulting features as we want. Hence, we can also use variable feature numbers depending on the view. If a view is easy to obtain for registration, but not so frequently appear in practice, then we can use a smaller number of features. More important views get bigger feature numbers.

In generating descriptors, training is considered as a step to create space basis and matrix transform for feature extraction and as mentioned in Section 5, many views are trained for one registered view to increase the retrieval ability and reliability. If we can embed more information in the step of training, the registration can be done with smaller information. For example, for the registered view $(30^\circ, 0^\circ)$, we use 9 surrounding views $(10^\circ, 0^\circ)$, $(20^\circ, 0^\circ)$, $(30^\circ, 0^\circ)$, $(40^\circ, 0^\circ)$, $(50^\circ, 0^\circ)$, $(30^\circ, -20^\circ)$, $(30^\circ, -10^\circ)$, $(30^\circ, 10^\circ)$, $(30^\circ, 20^\circ)$ for training. As summarized in Figure 6, for one view registration, the training is done with 6 to 9 views around the registered view. For this experiment, we have given three ways to extract features based on basic feature extraction method described in Section 4. Number of subregions and number of features on subregions vary. So for some views, 5 holistic features and 5 features for each of the five subregions are extracted which results in 30-dimensional view-specific feature vector, and for other views 5, holistic features and 2 features

for 5 subregions are extracted producing 15 dimensional vector. If a view is close to profile, we use 5 holistic features and 5 features for 2. For details for our experiment, see Figures 3 and 6. For one view, one image is selected.

In the experiment for multiview descriptor optimization with rendered images from 3D facial models of 108 individuals, half of the images were used for training and the other half were used for test. We show some examples of quasiview in Figure 7 which shows the influence of each represented view. Big yellow spots are the views for registration and small red spots indicate corresponding quasiviews with allowed error rate 0.05. Therefore, the region covered by small spots surrounding a big spot indicates the influence of the registered view (the big spot). For example, when we register the very front view (the left most one in the middle row in Figure 7), the horizontally 30°-rotated and vertically 20°-rotated views also could be recognized with error rate 0.05.

Through experiments with various combinations of quasiviews, a set of optimal views could be selected to create final multiview 3D descriptor. An example of such possible descriptor from the rendered images contains 13 views with 240-dimensional feature vector as shown in Figure 6. With the allowed error rate 0.05, this descriptor was able to retrieve the rendered images in the test database from 93.93% of the views in view mosaic of horizontal rotation $[-90^\circ \cdots 90^\circ]$ and vertical rotation $[-30^\circ \cdots 30^\circ]$. Figure 8 shows the covered region of the views by the selected 7 views (right half of the view space which corresponds to positive horizontal rotation) considering the symmetry of the horizontal rotation. Figure 9 shows an example which face views are needed for registration to recognize the face in almost any pose. The 7 views are to be registered to recognize a face in the 93.93% of the view space where horizontal rotation $[0^\circ \cdots 90^\circ]$, vertical rotation $[-30^\circ \cdots 30^\circ]$, and their combined rotation of a head are allowed. It means that we can retrieve a face in various poses within allowed error rate 0.05 when we register only 7 views in a condition that a given face is symmetric. For a reference, when we allowed error rate of 0.1, it covers 95.36% of the view space, 97.57% for error rate 0.15, and 97.98% for error rate 0.2. For the experiment, the testing views are situated at intervals 5 degrees while a 10-degree interval is used for training.

As a reference, the MPEG-7 AFR [9, 10] has 48 dimensions with error rate 0.3013 and 128 dimensions with error rate 0.2491 for photograph images. Here we used the error rate of ANMRR (average normalized modified retrieval rank), the MPEG-7 retrieval metric, which indicates how many of the correct images are retrieved as well as how highly they are ranked among the retrieved ones. Details about ANMRR can be found in MPEG related documents like [14].

7. CONCLUSION

We have shown how the single-view face descriptor could be extended to multiview one in efficient way by checking the size of quasiview, which is a measure of the view influence. For the experiment, the 3D facial mesh models of 108 subjects are used and their rendered images are used for training and test with 50/50 ratio. Only 13 views could be chosen as registered views throughout our optimization. This descriptor in 240 dimensions is able to retrieve images of 93.93% views of total region of view mosaic of horizontal rotation from -90° to 90° and vertical rotation from -30° to 30° within error rate 0.05.

The aim of this new descriptor is to be used to extract a face in any view by containing compact 3D information by optimization for how many and which views are to be registered. The extension to multiview is not very costly in terms of number of registration views thanks to the quasiview analysis. Even though we have used a specific face descriptor for the experiment, the potential of this method enables us to include any available 2D face recognition methods by showing how to combine them in optimized way by checking quasiview size. Ongoing research includes new feature extraction methods for profile views and missing view interpolation in the registration step.

REFERENCES

[1] A. Samal and P. A. Iyengar, "Automatic recognition and analysis of human faces and facial expressions: a survey," *Pattern Recognition*, vol. 25, no. 1, pp. 65–77, 1992.

[2] S. Z. Li, L. Zhu, Z. Q. Zhang, A. Blake, H. J. Zhang, and H. Shum, "Statistical learning of multi-view face detection," in *Proceedings of the 7th European Conference on Computer Vision (ECCV '02)*, vol. 4, pp. 67–81, Copenhagen, Denmark, May 2002.

[3] Y. Li, S. Gong, and H. Liddell, "Support vector regression and classification based multi-view facedetection and recognition," in *Proceedings of the 4th IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 300–305, Grenoble, France, March 2000.

[4] G. Shakhnarovich, L. Lee, and T. Darrell, "Integrated face and gait recognition from multiple views," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '01)*, vol. 1, pp. 439–446, Kauai, Hawaii, USA, December 2001.

[5] V. Blanz and T. Vetter, "Face recognition based on fitting a 3D morphable model," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 9, pp. 1063–1074, 2003.

[6] A. M. Bronstein, M. M. Bronstein, and R. Kimmel, "Expression-invariant 3D face recognition," in *Proceedings of the 4th International Conference on Audio- and Video-Based Biometric Person Authentication (AVBPA '03)*, vol. 2688 of *Lecture Notes in Computer Science*, pp. 62–69, Guildford, UK, June 2003.

[7] D. M. Gavrila and L. S. Davis, "3-D model-based tracking of humans in action: a multi-view approach," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '96)*, pp. 73–80, San Francisco, Calif, USA, June 1996.

[8] K. W. Bowyer, K. Chang, and P. Flynn, "A survey of approaches and challenges in 3D and multi-modal 3D + 2D face recognition," *Computer Vision and Image Understanding*, vol. 101, no. 1, pp. 1–15, 2006.

[9] A. Yamada and L. Cieplinski, "MPEG-7 Visual part of eXperimentation Model Version 17.1," ISO/IEC JTC1/SC29/WG11 M9502, Pattaya, Thailand, March 2003.

[10] T. Kamei, A. Yamada, H. Kim, W. Hwang, T.-K. Kim, and S. C. Kee, "CE report on Advanced Face Recognition Descriptor,"

ISO/IEC JTC1/SC29/WG11 M9178, Awaji, Japan, December 2002.

[11] W.-S. Lee and K.-A. Sohn, "Face recognition using computer-generated database," in *Proceedings of Computer Graphics International (CGI '04)*, pp. 561–568, IEEE Computer Society Press, Crete, Greece, June 2004.

[12] W.-S. Lee and K.-A. Sohn, "Database construction & recognition for multi-view face," in *Proceedings of the 6th IEEE International Conference on Automatic Face and Gesture Recognition (FGR '04)*, pp. 350–355, IEEE Computer Society Press, Seoul, Korea, May 2004.

[13] D. B. Graham and N. M. Allinson, "Characterizing virtual eigensignatures for general purpose face recognition," in *Face Recognition: From Theory to Applications*, H. Wechsler, P. J. Phillips, V. Bruce, F. Fogelman-Soulie, and T. S. Huang, Eds., pp. 446–456, Springer, Berlin, Germany, 1998.

[14] G. Park, Y. Baek, and H.-K. Lee, "A ranking algorithm using dynamic clustering for content-based image retrieval," in *Proceedings of the International Conference Image and Video Retrieval (CIVR '02)*, vol. 2383 of *Lecture Notes in Computer Science*, pp. 328–337, London, UK, July 2002.