ELSEVIER

Image and Vision Computing 18 (2000) 881–905

image AND vision COMPUTING

www.elsevier.com/locate/imavis

Expert system for automatic analysis of facial expressions

M. Pantic*, L.J.M. Rothkrantz

Faculty of Information Technology and Systems, Department of Knowledge Based Systems, Delft University of Technology, P.O. Box 356, 2600 AJ Delft, The Netherlands

Received 29 August 1998; received in revised form 30 June 1999; accepted 25 January 2000

Abstract

This paper discusses our expert system called Integrated System for Facial Expression Recognition (ISFER), which performs recognition and emotional classification of human facial expression from a still full-face image. The system consists of two major parts. The first one is the ISFER Workbench, which forms a framework for hybrid facial feature detection. Multiple feature detection techniques are applied in parallel. The redundant information is used to define unambiguous face geometry containing no missing or highly inaccurate data. The second part of the system is its inference engine called HERCULES, which converts low level face geometry into high level facial actions, and then this into highest level weighted emotion labels. © 2000 Elsevier Science B.V. All rights reserved.

Keywords: Intelligent multi-modal user interface; Hybrid facial feature extraction; Facial action tracking; Facial expression emotional classification

1. Introduction

The user interface for the computer systems is evolving into an intelligent multi-modal interface. It is now moving away from keyboard-given instructions to more natural modes of interaction, using visual, audio and sensorial means. This is the first step in achieving a human-like communication between man and machine.

Human communication has two main aspects: verbal (auditory) and non-verbal (visual). Words are the atomic information units of the verbal communication. Phenomena like facial expressions, body movements and physiological reactions are the atomic units of the non-verbal communication. Although it is quite clear that non-verbal gestures are not necessary for successful human interaction (e.g. phone calls), considerable research in social psychology has shown that non-verbal gestures can be used to synchronise dialogue, to signal comprehension or disagreement, to make dialogue smoother and with fewer interruptions [3,35]. This finding itself suggests that multi-media man-machine communication systems could promote more efficient performance.

At the moment there are several systems available for automatic speech recognition. On the other hand, a complete and accurate system for vision-based facial gesture analysis has not been developed yet. This triggered our interest in this topic.

1.1. Automation of non-verbal communication

Automated Systems for Non-verbal Communication is an ongoing project at the Knowledge Based Systems department of the TU Delft [38]. The goal of our project is the development of an intelligent automated system for the analysis of non-verbal communication. The system has to provide qualitative and quantitative information about different non-verbal signals at different levels. On the lowest level, the system should detect a non-verbal signal shown by the observed person. On the next level, the system should categorise the detected signal as a specific facial action (e.g. smile), a specific body action (e.g. shoulders shrug), a specific vocal reaction (e.g. high speech velocity), or a specific physiological reaction (e.g. sweating). On a higher level the system should give an appropriate, for example an emotional, interpretation of the recognised signal. On the highest level the system should reason about the intentions of the user.

Our intelligent analyser of human non-verbal communication should cope with the registration, processing and interpretation of the non-verbal communication signals. The multi-modal input to the system will consist of sound-, image-, and sensor data. Then analysis of recorded speech, facial expression, body movement, and physiological sensor data should be performed. These analyses have to be implemented as modules that can operate independently

* Corresponding author.
E-mail addresses: m.pantic@cs.tudelft.nl (M. Pantic); l.j.m.rothkrantz@cs.tudelft.nl (L.J.M. Rothkrantz).

PII: S0262-8856(00)00034-2

as well as the parts of a single system. The final result of the system will be a data-fusion of the results of the modules executed in parallel. This would represent the hypothesis about the intentions of the observed person.

First, we investigated automatic recognition and emotional classification of facial expressions in order to achieve such an automatic analyser of human non-verbal communication. This paper discusses the results of our research implemented as Integrated System for Facial Expression Recognition (ISFER).

1.2. Automatic facial expression analysis

Facial expressions play an essential role in human communication. As indicated by Mehrabian [24], in face-to-face human communication only 7% of the communicative message is due to linguistic language, 38% is due to paralanguage, while 55% of it is transferred by facial expressions. Therefore, in order to facilitate a more-friendly man–machine interface of new multimedia products, vision-based facial gesture analysis is being studied world wide in the last ten years. Numerous techniques have been proposed.

In their early work Kato et al. [18] used isodensity maps for the synthesis of facial expressions. They pointed out that the change of the extracted isodensity maps, which accompanies the change of a facial expression, could be effectively used for emotional classification of the facial expression. Still, this was not investigated in detail.

The work of Terzopoulos and Waters [36] also deals exclusively with the synthesis of facial expressions, rather than their interpretation. Their 3D dynamic face model combines a physically based model of facial tissue with an anatomically based facial muscle control process. In order to animate the human face with their face model, they use deformable curve techniques for estimating face muscle contraction parameters from video sequences.

Morishima et al. [26] reported on a 5-layered manual-input neural network used for recognition and synthesis of facial expressions. Zhao and Kearney [46] described singular emotional classification of facial expressions using a 3-layered manual-input backpropagation neural network. Kearney and McKenzie [19] developed a manual-input memory-based learning expert system, which interprets facial expressions in terms of emotion labels given by college students without formal instruction in emotion signals.

Another facial expression recognition system that requires a manual pre-processing is the system introduced by Kanade et al. [5]. Their system recognises Action Units (AUs) and AUs combinations [8] in facial image sequences using Hidden Markov Model. After manual marking of facial feature points around the contours of the eyebrows, eyes, nose and mouth in the first frame of image sequence, Kanade et al. use Lucas–Kanade optical flow algorithm [23] to track automatically the feature points in the remaining frames. In the case of the upper face, Wu–Kanade dense optical flow algorithm [43] and high gradient component detection is used to include a detailed information from the larger region of the forehead [22].

In 1997 and 1998, each of the four most influential research groups in the field of vision-based facial gesture analysis published a summary of their previous work.

Essa and Pentland [10] presented the results on recognition and singular emotional classification of facial expressions based on an optical flow method coupled with geometric, physical and motion-based face models. They used 2D motion energy and history templates that encode both, the magnitude and the direction of motion. By learning the “ideal” 2D motion views for four emotional expressions (anger, disgust, happiness and surprise), they defined spatio-temporal templates for those expressions. Although the approach proposed by Essa and Pentland has not been still fully validated, it should be noted that spatio-temporal templates of facial expressions form a unique method for facial expression emotional classification.

Black and Yacoob [2] also utilised an optical flow model of image motion for facial expression analysis. Their work explores the use of local parameterised optical flow models for the recognition of the six basic emotional expressions (sadness, happiness, anger, disgust, fear and surprise [7,9]).

Kobayashi and Hara [20] reported on real-time recognition, singular emotional classification and synthesis of the six basic emotional expressions. They worked on realisation of an animated 3D face-robot that can recognise and reproduce the emotional expressions. They use brightness distribution data of facial image and a 3-layered backpropagation neural network for classification and synthesis of facial expressions.

The researchers of MIRALab [37] reported on recognition, singular emotional classification and animation of human facial expressions. To construct a 3D virtual (cloned) face they use discrete snakes and two 2D point-based face templates. To reproduce the observed facial expression on the virtual face, a mapping is carried out from the tracked points of the face templates to 21, from a total of 65 minimal perceptible actions (similar to AUs of FACS [8]). A rule-based approach is employed to categorise the observed expression in one of the six basic emotional classes.

Each of the methods described above has some limitations. They either deal with the synthesis of facial expression which do not attempt to give an interpretation of it [18,36], or they give a low level of interpretation [5]. They use some semi-automatic or completely manual procedures for tracking facial features [5,19,26,46]. They either use AUs-coded description of the six basic emotional expressions that cannot be validated against the linguistic description given by Ekman [2,20,26], or do not explain the rules for emotional classification of expressions at all [37]. Except for JANUS [19], all of the described systems perform only singular emotional classification of facial

expressions. None of the approaches deals with the issue of blended emotional expression [9].

As explained in the next section, our approach to automatic facial expression analysis attempts to cope with the issues listed above.

1.3. Integrated system for facial expression recognition

The aim of our research is to design and implement a completely automated system for recognition and emotional classification of facial expressions. The project is still ongoing. At the moment, ISFER has the following functionality:

1. automatic extraction of the facial features from digitised facial images;
2. automatic encoding of face actions (described in terms of Action Units (AUs) [8]);
3. automatic classification of face actions in six basic emotion categories (happiness, anger, surprise, fear, sadness, and disgust [7,9]).

In contrast to the existing facial feature detectors (e.g. [10,18,20]) which utilise single image processing technique, ISFER represents a hybrid approach to facial feature detection. Each of the existing methods is based either on discrete snakes and template matching [37], or on optical flow models of image motion [2,5,10], or on 3D wireframe face models [36,37], or on brightness distribution data of facial image combined with a neural network approach [20]. Our approach to recognition of facial expression combines multiple feature detection techniques, which are applied in parallel. So instead of fine-tuning and completing existing facial feature detectors or inventing new ones, we propose to combine known techniques.

The motivation for combining detectors is the increase in quality of the combined detector. All feature-tracking algorithms have circumstances under which they perform extremely well. Also, they all have facial features that they can detect better. A combined detector will have less weak properties and perform better than the best single detector. Introducing redundancy by applying multiple detectors per facial feature and then choosing the best of the acquired results will finally yield in a more complete set of detected facial features. The ISFER Workbench has been implemented according to this multi-detector paradigm.

ISFER deals with a static face action. This means that only the end-state of the facial movement is measured in comparison to the neutral facial position. The movement itself is not measured. In addition, ISFER does not deal with a continuous tracing of someone's face. It deals with still facial images, not with image sequences. In other words, the system recognises and emotionally classifies stable patterns of facial expressions.

To avoid the problem of rigid head motions, that is to achieve successful acquisition of full-face and profile images, we explicitly specified the camera setting. Two digitised cameras should be mounted on two holders attached to a headphone-like device. One camera holder should be placed in front of the face (frontal-view) and the other on the right side of the face (side-view). By this, the cameras will move together with the head. Each facial view will be without any change in size and orientation of the face compared to the previously acquired images.

Current expression classifiers [2,10,20,37] have the limitation of categorising the examined expressions exclusively into one of the emotion categories. In turn, they are not capable of performing recognition and classification of non-prototypic expressions (such as a blend of emotional expressions [7,9]). To overcome this problem and develop a system that can recognise complex non-prototypic expressions, face actions should be recognised. Except for JANUS [19], which is partially a manual system, and the vision-based system introduced by Kanade et al. [5], which does not deal with emotional classification of facial expressions, none of the existing systems deals with the recognition of face actions. ISFER represents a completely automated system that has been developed to convert the face geometry (localised facial features) into a description of face actions, and then this into weighted emotion labels.

The reasoning of the system is person-independent. This means that the process of facial expression recognition and emotional classification does not depend on physiognomic variability of the observed persons. The generic face model, described further in the text, facilitates person-independence of ISFER reasoning.

ISFER consists of three major parts (see Fig. 1), namely image data extraction, data evaluation, and data analysis. The first part of the system is the ISFER Workbench, which represents a collection of facial feature detection algorithms. The second part of the system is the Facial Data Evaluator, which represents a connection between the data generator and the inference engine of the system. This part of the system makes a best possible selection from the redundantly detected facial features so that the resulting face geometry does not contain missing and highly inaccurate data. The obtained face geometry forms the input to the system's reasoning mechanism called HERCULES.

Dealing with ambiguous information encountered in the examined facial expression is partially based on the knowledge about the neutral facial expression. The first step in performing automatic analysis of someone's facial expression is, therefore, the analysis of his/her neutral facial expression. To ensure correct extraction of the facial features from someone's neutral facial expression, it is highly recommended that the results of automatic feature detection are visually inspected and if necessary, that the choice of facial feature detectors is further manually made. Analysis of each next expression of the observed person is performed in a completely automatic way.

The theoretical background of face action recognition and facial expression emotional classification is given in Section 2. Our face model is explained in Section 3. The framework

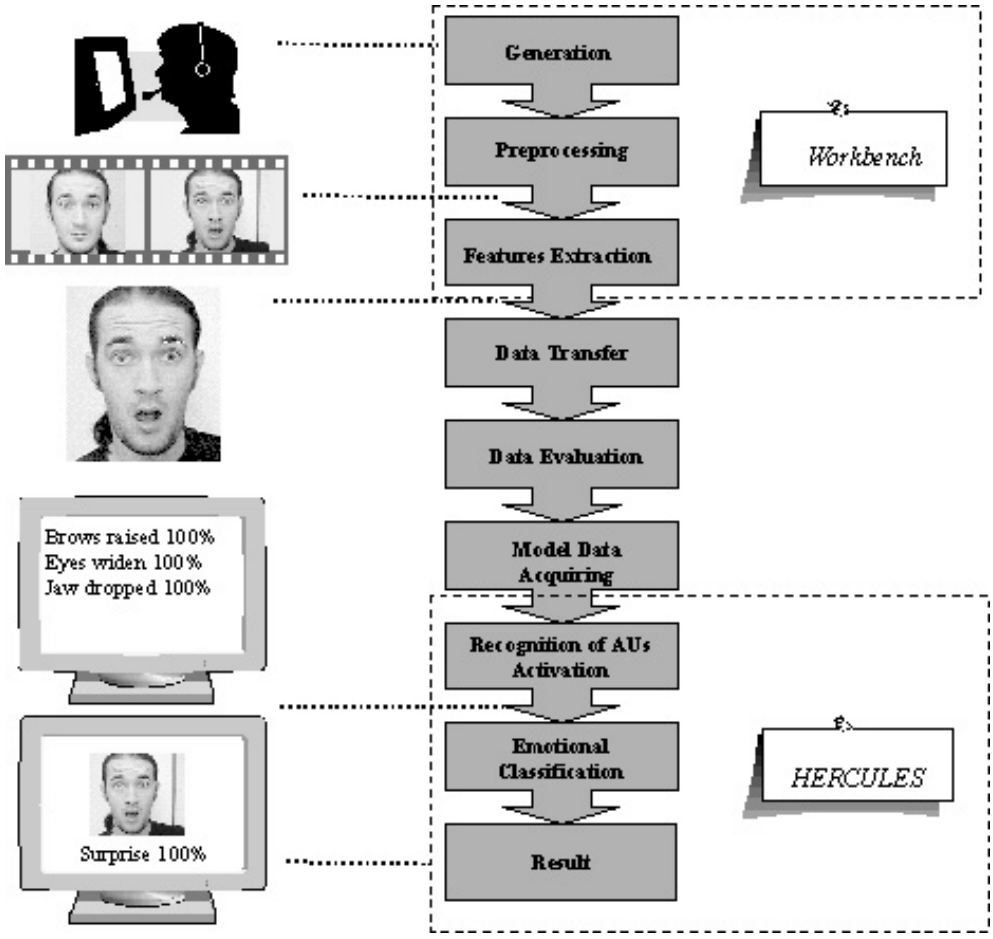


Fig. 1. ISFER structure.

for hybrid facial feature detection is explained in Section 4. Section 5 describes dealing with ambiguous facial expression information. The process of facial data analysis is explained in Section 6. The concluding remarks are given in Section 7.

2. Theoretical background

One of the fundamental issues about the recognition of prototypic emotional facial expressions is the categorisation of the visual information that the examined faces might reveal [4]. According to Yamada [44], this process of categorisation has two stages. The first one is to define the set of categories we have to deal with in classifying expressions. The second one is to define the mechanism of categorisation.

Most psychological studies on facial expression classification have been related to the first step. Probably the most known and the most commonly used study on classification of facial expressions is the cross-cultural study on existence of *universal categories of emotional expressions* [7,9,12,16]. Ekman defined six such categories, called *six basic emotions*: happiness, sadness, surprise, fear, anger and disgust [7]. He described each of the basic emotions in terms of a facial expression that universally and uniquely characterises that emotion. Although some psychologists like Russell [33] doubt the universality of the six basic emotions, most of the researches of vision-based facial gesture analysis rely on Ekman's emotional categorisation of facial expressions [2,20,37]. The production rules of ISFER inference engine, which perform emotional classification of facial expressions, are also based on Ekman's description of the six basic emotions.

In contrast to the first stage of the process of expression categorisation, there have been rather few studies on the second stage. So, which kind of information from the face we use in order to classify a certain facial expression into a particular emotional category is still an open question [4]. Probably the most known study on the subject is FACS [8].

Facial expressions represent a visible consequence of facial muscle activity. FACS represents a system that describes facial expressions in terms of codes of this facial muscle involvement. The activation of the muscles is described by the system as the activation of 44 Action Units (AUs). Each AU corresponds with the contraction produced by one or a group of related muscles. Activation of an AU is described in terms of facial appearance change,

i.e. change of the facial features such as eyebrows, eyes and mouth caused by activity of the underlying muscle(s). With FACS all visually distinguishable facial movements can be described as AUs-codes.

On the other hand, FACS has been developed for human observers. So, neither facial muscle activity nor AUs-codes can be extracted directly from a digital image. What is necessary for resolving of this problem is to define automatically extractable visual properties of facial expressions. The results of Johansson's point-light display experiments gave a clue to this problem.

Bassili [1], and later Bruce [4], requested the stimulus person to make various facial expressions while having white marks placed on the face at random. The subjects observed only the movements of the white marks through a monitor. They were quickly aware of seeing a face and they could easily say what kind of facial expression the movement of the white marks represented. Johansson's point-light display experiments suggest that the visual properties of the face, regarding information on facial expressions, could be made clear by describing the movements of points belonging to the facial features (eyebrows, eyes, nose, mouth, and chin) and then by analysing the relationships between those movements.

This triggered the researchers of vision-based facial gesture analysis to make different attempts to determine point-based visual properties of facial expressions. This concerns defining some point-based face model (e.g. [19,20,26,37]), defining a mechanism for automatic extraction of these points from digital facial image, and establishing a relation between the movement of the extracted points and the AUs. Then some AUs-coded description of the six basic emotional expressions is used to categorise AUs-coded description of the shown facial expression.

We followed the same process when defining automatically extractable visual properties of facial expressions. Our face model as well as the relationship between the face model and the AUs is defined in the next section. The mechanism for automatic extraction of the model features is described in Section 4. The used AUs-coded description of the six basic emotions and the mechanism for emotional classification of facial expressions are explained in Section 6.

3. Face model

Currently, facial expression recognition systems use either complicated 3D wireframe face models [36,37] or consider only averaged optical flow within local regions (e.g. forehead, eyes, and mouth) [2]. Using currently available vision techniques, it is difficult to design a 3D face model that accurately represents facial geometric properties. The initial adjustment between the 3D wireframe and the surface images is usually manual, which affects the accuracy of the recognition results. Similarly, accurate and dense information on facial expression get lost when only averaged optical flow within local facial regions is estimated.

There are also several existing 2D face models. An example is the Facial Landmarks model [19], which is not suitable for an automatic extraction of facial points. Another example is the model of 18 facial characteristic points proposed by Kobayashi and Hara [20]. In their face model none of the points belongs to the lower eyelid or to the upper lip. Facial movements, however, do effect displacement of both, the lower eyelid and the upper lip. The model of 22 facial points used by Morishima et al. [25] is the same as our frontal-view face model except for the points of the eyebrows. They use the centre of both, the lower eyebrow border and the upper eyebrow border, which form redundant information when used together.

We choose to define our face model as a point-based model composed of two 2D facial views, namely the frontal- and the side view. There are a number of motivations for this choice. As shown by Bassili [1] and Bruce [4], a point-based graphical face model resembles the model used by human observers when judging a facial expression. Consequently, expression-classification rules used by human observers (e.g. the rules of FACS) can be converted straightforwardly into the rules of an automatic classifier based on a point-based face model. Another motivation is the simplicity of validating a point-based face model. The changes in the position of the points in the face model are directly observable. By comparing the changes in the model and the changes in the modelled expression, the validity of the model can be visually inspected. Finally, combining a dual facial view into a single model yields a more realistic representation of 3D face and avoids inaccuracy and manual initialisation of a 3D wire-frame model.

The frontal-view model and the side-view model, considered separately, do not contain redundant information about the facial features. When coupled together, however, two facial views reveal redundant information about facial expression. Depending on success of the facial feature detection algorithms, this redundancy is used further to encode unambiguously the facial geometry in terms of AUs-codes (see Section 3.3 and Section 5).

Our model has the following characteristics.

1. The features defined by the model are extracted automatically from the still full-face images in the case of frontal-view and from the still profile images in the case of side-view.
2. The deformations of the features defined by the frontal-view model reveal changes in the appearance of eyes, eyebrows, nose, mouth and chin. The deformations of the features defined by the side-view model reveal changes in the appearance of forehead, nose, mouth, jaw and chin.
3. It is possible to establish a simple and unique relation between changes of the model features and separate AUs.

The frontal- and the side-view face model, as well as the

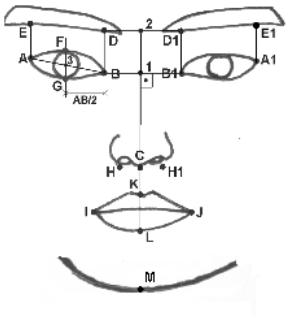


Fig. 2. Facial points of the frontal-view.

description of the AUs in terms of the model features, are described in the following two sections.

3.1. Frontal-view face model

The frontal-view face model is composed of 30 features that can be divided into two groups. The first group is formed by 25 features, which are defined in correspondence with a set of 19 facial points. These points are illustrated in Fig. 2 and described in Table 1. The features of the first group are given in Table 2. The second group is formed by five features, which represent four specific shapes of the mouth and one specific shape of the chin. Those features are described in Table 3, Figs. 3–6. Automatic extraction of the facial features defined by the frontal-view model is described in Section 4.

The frontal-view model has been generated and then validated through analysis, and respectively synthesis, of linguistic labels used to describe the visual properties of AUs [8]. For example, the analysis of the label *upward pull of the inner portion of the eyebrows*, which describes activation of AU1, caused the addition of the features f1 and f2 to the model. An observed increase of f1 and f2 will cause trained FACS coders [8] to conclude that AU1 is activated.

From a total of 44 AUs defined in FACS, 27 AUs can be uniquely described using our frontal-view face model. The importance of a unique representation of AUs-codes, and our way of achieving it in terms of our face model, can be explained using an example. In FACS, the activation of AU9 as well as the activation of AU10 is described with the label *upward pull of the upper lip*. It is also stated, however, that activation of AU9 obscures the activation of AU10. On the other hand, the label *wrinkled root of the nose* describes AU9 exclusively. To obtain uniquely defined AU10-code with our model, we are describing it as decreased f11 and non-increased f20.

FACS description of AUs and the representation of AUs-codes, using an informal reader-oriented pseudo-code, are given in Table 4.

Table 1
Facial points of the frontal-view

Point	Point description
B	Left eye inner corner, stable point
B1	Right eye inner corner, stable point
A	Left eye outer corner, stable point
A1	Right eye outer corner, stable point
H	Left nostril centre, non-stable
H1	Right nostril centre, non-stable
D	Left eyebrow inner corner, non-stable
D1	Right eyebrow inner corner, non-stable
E	Left eyebrow outer corner, non-stable
E1	Right eyebrow outer corner, non-stable
F	Top of the left eye, non-stable
F1	Top of the right eye, non-stable
G	Bottom of the left eye, non-stable
G1	Bottom of the right eye, non-stable
K	Top of the upper lip, non-stable
L	Bottom of the lower lip, non-stable
I	Left corner of the mouth, non-stable
J	Right corner of the mouth, non-stable
M	Tip of the chin, non-stable

Table 2
First group of the features of the frontal-view model

Feature	Feature description
f1	Angle ∠BAD
f2	Angle ∠B1A1D1
f3	Distance AE
f4	Distance A1E1
f5	Distance 3F, 3 is the centre of AB
f6	Distance 4F1, 4 is the centre of A1B1
f7	Distance 3G
f8	Distance 4G1
f9	Distance FG
f10	Distance F1G1
f11	Distance CK, C is 0.5HH1 (f0)
f12	Distance IB
f13	Distance JB1
f14	Distance CI
f15	Distance CJ
f16	Distance IJ
f17	Distance KL
f18	Distance CM
f19	Image intensity in circle (r(0.5BB1), C(2)) above line (D, D1)
f20	Image intensity in circle (r(0.5BB1), C(2)) below line (D, D1)
f21	Image intensity in circle (r(0.5AB), C(A)) left from line (A, E)
f22	Image intensity in circle (r(0.5A1B1), C(A1)) right from line (A1, E1)
f23	Image intensity in the left half of the circle (r(0.5BB1), C(I))
f24	Image intensity in the right half of the circle (r(0.5BB1), C(J))
f25	Brightness distribution along the line (K, L)

Table 3
Second group of the features of the frontal-view model

Feature	Feature description
f26	Lower lip shape shown in Fig. 3
f27	Mouth shape shown in Fig. 4
f28	Mouth shape shown in Fig. 5
f29	Circular shape of the furrows on the chin shown in Fig. 6
f30	Mouth shape when the upper lip is sucked in (mirrored shape of that shown in Fig. 4)

Fig. 3. Shape of the lower lip when pulled downwards.

Fig. 4. Mouth shape when lower lip is sucked in.

3.2. Side-view face model

The side-view face model is composed of 10 face profile points. The profile points correspond with the peaks and the valleys of the curvature of the profile contour function. Transforming the contour into a function graph offers the possibility to use mathematical tools such as MATLAB to locate automatically maximum and minimum points of extreme curvature [42]. The profile points are illustrated in Fig. 7 and described in Table 5. The features of the side-view face model are given in Table 6.

All of the profile points are located as extremities of the curvature of the profile contour function (Table 5). Usually, many extremities are found. By using a priori knowledge, we delete false positive/negative extremities (see description of the Find Profile Contour module, Table 10). The order of the selected extremities, however, can be changed if the tongue is visible. In that case a valley representing the attachment of the upper lip with the tongue, a peak representing the tip of the tongue, and a valley representing the attachment of the tongue with the bottom lip, would be detectable between the points P6 and P8. In the case of the lips sucked into the mouth, only the valley of P7 would be detectable while peaks P6 and P8 would not exist. Therefore it is important to track the profile points in a particular order. The points P1 to P5 should be located first. Then the points P10 and P9 should be located. After excluding all of the extreme cases, such as visible tongue, the points P8, P7 and P6 should be located.

Fig. 5. Mouth shape when cheeks are sucked in.

Fig. 6. Shape of the chin furrows when chin is raised.

Table 4
Representation of AUs with our frontal-view model

AU	FACS description	Mapped on model
1	Raised inner brows	increased f1 & f2
2	Raised outer brow	increased f1 or f2
4	Lowered brows or Frowned brows	non-increased f20, (decreased f1 & f2) or increased f19
5	Raised upper lid	increased f5 or f6
6	Raised cheek	increased f21 or f22
7	Raised lower lid	non-increased f20, non-increased f21, non-increased f22, $f9 > 0$, $f10 > 0$, $f5 > 0$, $f6 > 0$, decreased f7 or f8
8	Lips towards each other (teeth visible, lips tensed and less visible)	decreased f25, increased f11, $f17 > 0$
9	Wrinkled nose	increased f20
10	Raised upper lip	non-increased f20, decreased f11
12	Mouth corners pulled up	decreased f12, decreased f13, increased f14, increased f15
13	Mouth corners pulled sharply up	decreased f12, decreased f13, decreased f14, decreased f15
14	Mouth corner pulled inwards	increased f23 or f24
15	Mouth corner pulled downwards	increased f12 or f13
16	Depressed lower lip (see Fig. 4)	present f26
17	Raised chin (see Fig. 7)	present f29
18	Lips puckered (as pronouncing the word “fool”)	decreased f16, absent f28
20	Mouth stretched	increased f16, non-increased f12, non-increased f13
23	Lips tightened but not pressed	absent f27 & f30, decreased f17, $f17 > 0$, non-decreased f25, non-decreased f16, non-increased f12, non-increased f13
24	Lips pressed together	absent f27 & f30, decreased f17, $f17 > 0$, non-decreased f25, decreased f16, absent f28
25	Lips parted	threshold $> f18 > 0$ increased f17, or non-increased f18, increased f17, non-decreased f25
26	Jaw dropped	f18 between two thresholds
27	Mouth stretched	$f18 >$ threshold
28	Lips sucked in	$f17 = 0$
28b	Bottom lip sucked	Present f27
28t	Top lip sucked in	Present f30
35	Cheeks sucked in	Present f28
38	Nostrils widened	absent AUs: 8, 9, 10, 12, 13, 14, 15, 18, 20, 24, 28, increased HH1
39	nostrils compressed	decreased HH1
41	Lid dropped	non-decreased f7, decreased f9, decreased f5, or decreased f10, decreased f6, non-decreased f8

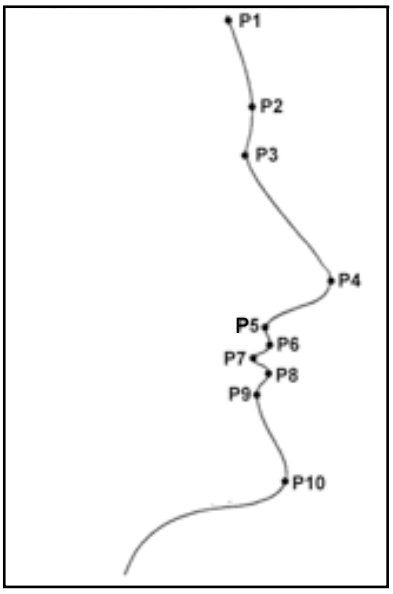


Fig. 7. Face profile points.

From a total of 44 AUs defined in FACS, 20 AUs can be uniquely described using our side-view face model. The representation of AUs-codes with our side-view face model is given in Table 6.

Harmon has developed a similar model of the profile points for a face identification system [15]. However, facial expression emotional classification and face identification are different tasks. Personal characteristics such as the length of the nose are considered as unimportant data in facial expression categorisation as well as the opening of the mouth is considered as noise in face identification. Our side-view face model is developed to be suitable for facial expression emotional classification and, therefore, merely resembles Harmon's model.

Table 5
Facial points of the side-view

Point	Point description
P1	Top of the forehead, joint point of the hair and the forehead
P2	Eyebrow arcade, first peak of the curvature of the contour function
P3	Root of the nose, first valley of the curvature of the contour function
P4	Tip of the nose, absolute maximum of the curvature of the contour function
P5	Upper jaw, first valley after the P4 peak of the curvature of the contour f-on
P6	Upper lip, first peak after the P4 peak of the curvature of the contour f-on
P7	Lips attachment, first valley after the upper lip (the P6 peak)
P8	Lower lip, first peak after the tip of the chin (the P10 peak)
P9	Lower jaw, first valley after the tip of the chin (the P10 peak)
P10	Tip of the chin, first peak starting from the end of curvature of the contour f-on

Table 6
Representation of AUs with our side-view model

AU	Movement of profile points
1	P2 upwards, curvature between P1 and P2 contains a row of peaks and valleys
1 + 2	P2 upwards, curvature between P1 and P2 remains without local extremities
4	P2 downwards, curvature between P2 and P3 is not increased
8	Distance P5-P6 increased, P6 outwards, P8 outwards, curvature between P6 and P8 is more straight and angular ([), distance P8-P10 increased
9	Curvature between P2 and P3 increased
10	P6 upwards and outwards, distance P5-P6 decreased, curvature between P2 and P3 is not increased
12	Distance P5-P6 decreased, P6 inwards, P8 inwards, distance P6-P8 increased
13	Distance P5-P6 decreased, P6 inwards, P8 inwards, distance P6-P8 remains same
15	Distance P5-P6 increased, curvature between P5 and P6 is not increased, P6 downwards, P8 downwards, distance P6-P8 not decreased
16	P8 downwards and outwards, distance P8-P10 decreased
17	P10 inwards
18	P6 outwards, P8 outwards, curvature between P6 and P8 is not [
19	Tongue showed, curvature between P6 and P8 contains two valleys and a peak
20	Distance P5-P6 increased, curvature between P5 and P6 is not increased, P6 inwards, P8 inwards, distance P6-P8 not decreased
23	Distance P5-P6 increased, curvature between P5 and P6 is not increased, P6 downwards and inwards, P8 upwards and inwards, distance P6-P8 decreased but it is >0 and >threshold_1
24	Distance P5-P6 increased, curvature between P5 and P6 is not increased, P6 downwards and inwards, P8 upwards and inwards, distance P6-P8 decreased but it is >0 and <threshold_1
25	Distance P6-P8 is increased, distance P4-P10 < threshold_2
26	Distance P4-P10 between threshold_2 and threshold_3
27	Distance P4-P10 > threshold_3
28	Points P6 and P8 are absent
28b	Point P8 is absent
28t	Point P6 is absent
36t	Bulge above the upper lip produced by the tongue, curvature between P5 and P6 is increased
36b	Bulge under the lower lip produced by the tongue, P9 is absent

3.3. Combined face model

The motivation for combining the frontal- and the side-view model is the increase in quality of the face model. With the frontal- and the side-view model separately, we can uniquely describe the activation of 27 and 20 different AUs respectively. When the two views are combined in a single face model, 29 different AUs can be uniquely described (see Table 7).

Also, each facial view is more suitable for observing some AUs activation. For example, the frontal-view model is more suitable for the description of the AUs that effect the eyes, while the side-view model is more suitable for the description of the AUs that effect the jaw and the chin. Furthermore, it is wiser to use the AUs description that does not depend on tracking some noisy image feature. For example, the curvatures of the profile contour function are

Table 7
Representation of AUs with the combined face model

AU	Recognition
1	Based on frontal-view model
2	Based on frontal-view model
4	Based on side-view model
5	Based on frontal-view model
6	Based on frontal-view model
7	Based on frontal-view model
8	Based on side-view model
9	Based on side-view model
10	Based on side-view model
12	Based on frontal-view model
13	Based on frontal-view model
14	Based on frontal-view model
15	Based on frontal-view model
16	Based on side-view model
17	Based on side-view model
18	Based on frontal-view model
19	Based on side-view model
20	Based on frontal-view model
23	Based on frontal-view model
24	Based on frontal-view model
25	Based on side-view model
26	Based on side-view model
27	Based on side-view model
28	Based on side-view model
28b	Based on side-view model
28t	Based on side-view model
35	Based on frontal-view model
36t	Based on side-view model
36b	Based on side-view model
38	Based on frontal-view model
39	Based on frontal-view model
41	Based on frontal-view model

less noisy than the image intensity that increases when wrinkles are present (or shadows, birthmarks and obstacles). Consequently, activation of AU4, AU9 and AU17 is detected more reliably from the side-view than from the frontal-view. Combining the two views yields a face model with fewer weaknesses and fewer complicated AUs-descriptions than a single-view model.

When processed in parallel, the two facial views reveal redundant information about the facial expression. The way of reducing this redundancy depends on a twofold, namely the successfulness of localising the model features and the suitability of describing an AU with the particular facial view model. Analysing the suitability of each facial view model for the description of a particular AU-code, resulted in the rules of Table 7. In the case that all of the model features are successfully tracked the rules of Table 7 will be applied for a final generation of face geometry. In the case that the contour of the profile is not successfully detected, the facial expression will be AUs-encoded using the rules given in Tables 4 and 5. If some frontal-view model feature is not located successfully, the related rules of Table 7 will be substituted with the appropriate rules of Table 6. A detailed description of genera-ting unambiguous face geometry is given in Section 5.

4. Hybrid facial feature detection

In contrast to the existing facial feature detectors that utilise a single image processing technique ([2,5,10,18,20]), ISFER represents a hybrid approach to facial feature detection. The ISFER Workbench, which represents the first part of our system (see Fig. 1), combines multiple feature detection techniques that are applied in parallel. Instead of fine-tuning the existing facial feature detectors, we are combining known techniques. The ISFER Workbench is a Java-implemented tool that has been designed according to this multi-detector paradigm. The overall structural design of the ISFER Workbench and its GUI are explained in Rothkrantz et al. [32]. We are giving an overview of the workbench design in Section 4.1 where we include a description of all facial feature detectors integrated into the ISFER Workbench. Sections 4.2 and 4.3 provide a detailed description of newly developed facial feature detectors, namely the NN-based eye detector and the fuzzy classifier of mouth expressions.

4.1. ISFER workbench

The structure of the ISFER Workbench can be illustrated as shown in Fig. 8. The modules of the ISFER Workbench can be classified into three groups. The modules for generating digital dual-view face images, for filtering the image data and for feeding other modules with this information belong to the pre-processing group. The modules of the pre-processing group are explained in Table 8. The modules that perform detection of facial regions (e.g. mouth region) belong to the detection group. The modules of the detection group are also described in Table 8. The modules that localise the facial features belong to the extraction group. The modules that perform tracking of the upper face features (eyebrows, eyes and nose) are given in Table 10. The modules that localise the mouth and the profile are described in Table 10.

The ISFER Workbench operates in two modes, namely, single-detector mode and multi-detector mode. In the single-detector mode the user can select and then connect an arbitrary number of modules in order to form a network of modules that performs a desired detection of the facial features (e.g. Fig. 9). At any moment the current network is displayed to the user in a form of a directed graph where the nodes of the graph depict the modules and the branches of the graph depict the connections between the modules. Two modules can be connected in a network when the output of one module forms the input to the other module. Each time when a connection is made, it will be checked if the data types of the modules match. Only if they match, the connection between the modules will be allowed.

The workbench modules used in the example of Fig. 9 are not the only algorithms used for localising the eyes, eyebrows and mouth. For each facial feature, several detectors have been integrated into the ISFER Workbench (see

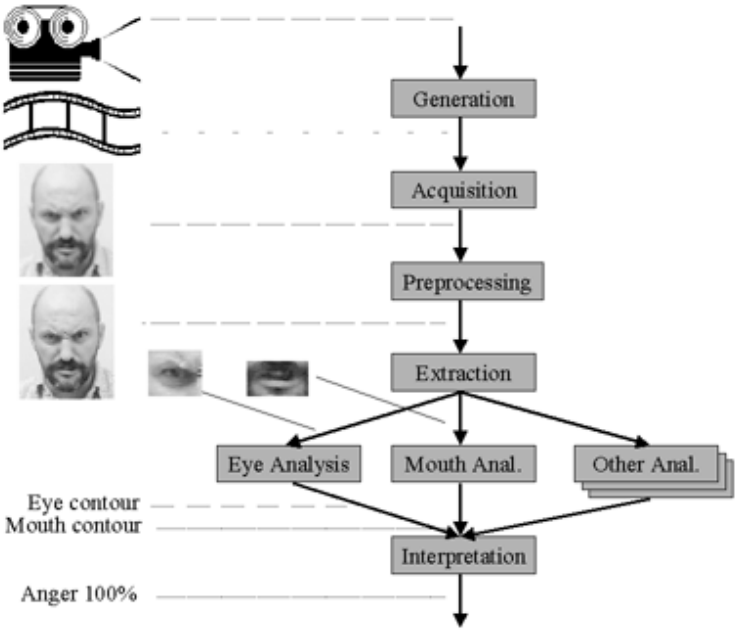


Fig. 8. Algorithmic representation of the ISFER workbench.

Tables 9 and 10). In the multi-detector mode, the user does not interact with the workbench; each and every facial feature detector integrated into the workbench is invoked automatically. The result of each detector is stored in a separate file. Those files form the input to the next stage of the system's processing—processing of redundant, missing and inaccurate data that will result finally in unambiguous face geometry. The Facial Data Evaluator is described in Section 5.

4.2. Neural network approach to eye detection

Neural networks have an excellent capability to recognise specific patterns. This property is exploited here to extract graphical patterns from digital images. In our case the graphical patterns that we are searching for are combinations of pixel values (grey values). Due to its robustness and generalisation power, a neural network can also recognise patterns that resemble the original pattern. This is in contrast to the conventional image processing techniques, which are usually not capable of performing such an approximation.

The eye detector, implemented as the Eye NN workbench module, utilises a $81 \times 4 \times 1$ backpropagation neural network with a sigmoid transfer function. To detect the eyes in a digitised image (320×240 pixels; 256 grey levels), the detector processes in two stages, coarse and fine.

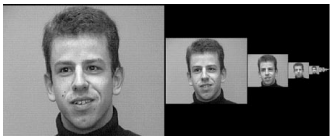
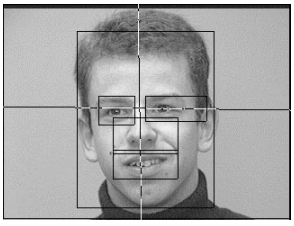
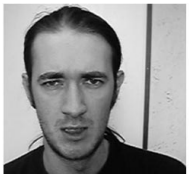
For each of the eyes, a 9×9 pixels box enclosing just the eye is located in the coarse stage. The eye region is first segmented from the input image using the workbench module MRP to RFM. Then, a 9×9 pixels scan window is scanned over the obtained eye region. Each pixel of the scan window is attached to an input neuron of the neural network, which has been trained to recognise the iris of the eye. The location where the highest neural response has been reached is assumed to be the centre of the iris. In the next step, the scan window is set around this point. If the location where the highest neural response has been reached remains the same as in the previous step, the position of the iris is found. Otherwise, this step is repeated until the iris is found. A 9×9 pixels scan window that will be used in the fine stage of the algorithm is then set around the iris.

In the fine stage, the eye sub-features are located. The idea about searching the characteristic points of the eye by applying a neural network originates from the Hierarchical Perceptron Feature Location method of Vincent et al. [39]. A difference between the two methods is in the choice of the eye micro-features. The micro-features located by our eye-detector are illustrated in Fig. 10. These micro-features are invariant with respect to the size of the eye, with respect to the shown facial expression, and with respect to the person whom the eye belongs to. In correspondence with the iris of the eye (located in the coarse stage of the algorithm), a 9×9 pixels search area is set for each micro feature. Each pixel of a search area is attached to an input neuron of a neural network that has been trained to recognise the related micro-feature. The location of the highest neural response that is reached represents the location of the micro-feature. A priori knowledge such as the symmetrical position of the features is used to discard false positives.

The micro-features lie in fact on the border between the eyelids and the eye. The algorithm ends by approximating this border by two third-degree curves drawn through the locations of the micro-features (see Fig. 10).

For the experiments 252 full-face images of nine different persons were used. The pictures were divided into two groups of 126 images. Each group consists of two times

Table 8
The modules of the ISFER Workbench belonging to the pre-processing group and the detection group

Module	Module description
Image to colour (pre-processing)	Conversion of Java *Image* data to a flat array of pixels
Colour to grey (pre-processing)	Conversion of the colour picture to a grey picture
Convolution filter (pre-processing)	Noise removal and smoothing of the image by applying linear convolution filtering with Gaussian or a Uniform filter [13].
Median filter (pre-processing)	Enhancement of the continuous areas of constant brightness in the image and slight sharpening of the edges by applying non-linear Median filter [13].
Grey to MRP implemented in C (detection)	Creation of the layers of the Multi Resolution Pyramid by calculating the half of the current image resolution (rounded to higher integer value) and averaging squares of 2×2 to one, which half the image in both directions. The routine is performed recursively until both image sizes equal 1 (see the illustration hereunder).
MRP to RFM implemented in C (detection)  	The module reads the given Multi Resolution Pyramid and locates, on the given layer (the default layer is 2), the Raw Feature Map that represents a rough approximation of the locations of the facial features. First the head is located by applying sequentially the analysis of the vertical histogram (showing the colour-differences between the successive rows, pixel-wise) and then the horizontal histogram (showing the colour-differences between the successive columns, pixel-wise). The peaks of the vertical histogram of the head box correspond with the border between the hair and the forehead, the eyes, the nostrils, the mouth and the boundary between the chin and the neck. The horizontal line going through the eyes goes through the local maximum of the second peak. The x co-ordinate of the vertical line going between the eyes and through the nose is chosen as the absolute minimum of the contrast differences found along the horizontal line going through the eyes. The box bounding the left eye is first defined to have the same size as the upper left face quadrant (defined by the horizontal and the vertical line) and to lie so that the horizontal line divides it in two. By performing the analysis of the vertical and the horizontal histogram, the box is reduced so that it contains just the local maximums of the histograms. The same procedure is applied to define the box that bounds the right eye. The initial box bounding the mouth is set around the horizontal line going through the mouth, under the horizontal line going through the nostrils and above the horizontal line representing the border between the chin and the neck. The initial box bounding the nose is set around the horizontal line going through the nostrils, under the horizontal line going through the eyes and above the horizontal line going through the mouth. By analysing the vertical and the horizontal histogram of an initial box, the box is reduced to contain just the tracked facial feature.
Find Head Contour, implemented in C (detection)  	The algorithm is based on the HSV colour model. The first step is to define the value of the parameter $Hue \in [-60,300]$. Analysis of 120 full-face images of different people results in the conclusion that the *Hue* of the face colour seldom exceeds the interval of $[-40,60]$. These experimental results also yield the fact that the range of *Hue* never exceeds 40 for the images of a single face, irrespective of change in the lightning conditions. The *Hue* is defined as $[-40 < averageHue - 20, averageHue + 20 < 60]$ where the average *Hue* is calculated as the average of the *Hue* in the box containing a horizontal middle of the face. The box is defined by analysing the vertical and the horizontal histogram of the input image. The face is then extracted as the biggest object in the scene having the *Hue* in the defined range. A similar method, but more general across the human specie and based on the relative RGB model, is presented by Yang and Waibel [45].

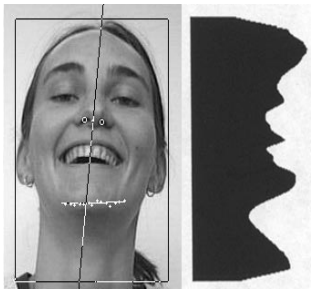
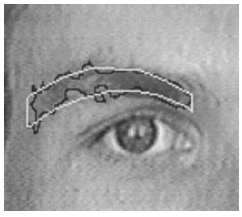
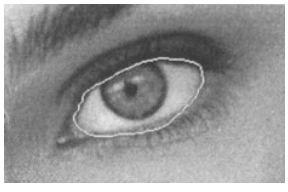
seven basic emotional expressions (happiness, anger, fear, surprise, disgust, sadness, and neutral) shown by nine different persons. One group of images has been used as a training set and the other as the testing set of images. The images were taken at a resolution of 320×240 pixels and a colour depth of 24 bits, which was reduced to 256 grey levels.

First, all of the images have been given to a human observer. Using Adobe Photoshop and a mouse device, the observer pointed the exact location of the eye micro-features. Per image and for each micro-feature the training pattern has been obtained by extracting (row by row) 81-dimensional vector of the grey levels of the pixels in 9×9 pixels window, which has been set around the micro-feature pointed out by the user. Per micro-feature a $81 \times 4 \times 1$ backpropagation neural network has been trained. Each network was trained using 126 input vectors until a small mean squared error (<0.01) was reached for the training vectors (after approximately 1000 training epochs).

The performance of the detection scheme is evaluated by calculating the block distance (maximal difference in x and y directions) between the estimated micro-features and the manually located micro-features. If we denote the estimated position of a micro-feature by $p = (x, y)$ and the manually assigned point by $p_M = (x_M, y_M)$ then the performance indicator is expressed by $d(p, p_M)$.

The performance of the algorithm has been measured first for the training set of images. The results of this test are shown in Table 11. From this table one can see that the localisation error, for all of the micro-features in all 126

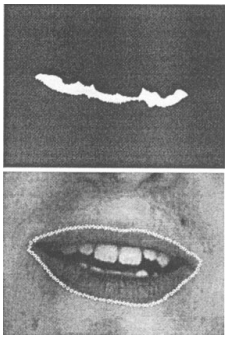
Table 9
The modules of the ISFER Workbench integrated in the extraction group—Nose-, eyes- and eyebrows detectors

Module	Module description
Find Nose & Chin, implemented in C (extraction) 	First the head is segmented from the facial image by analysing the horizontal and the vertical signature of the image [14]. Then, the linearly filtered and clipped image is thresholded and the seed-fill algorithm is applied [6] for colouring of the important facial regions such as eyes, nostrils and mouth. The symmetry line between the important facial regions is found using an adapted version of the algorithm based on Vornoi diagrams and presented by O'Rourke [27]. The region, where it is looked for the nostrils, is defined in correspondence with the second deepest valley of the brightness distribution along the symmetry line (a similar algorithm has been used by Hara and Kobayashi [20]). The important facial regions, which are found previously by the seed-fill algorithm, which belongs to the nostrils region, which are at approximately the same perpendicular distance from the symmetry line, and which have the highest intensity values, are located as the nostrils. The tip of the chin is located as the first peak after the third deepest valley (the mouth) of the brightness distribution along the symmetry line.
Curve fitting of the eyebrow, simplify polygon, draw polygon implemented in C (extraction) 	To localise the left eyebrow, the upper left face quadrant is first segmented from the facial image using the facial axis found by the module MRP to RFM and the contour of the face found by the module Find Head Contour. The eye-eyebrow region is determined throughout analysing the horizontal and the vertical signature of the linearly filtered and thresholded image of the upper left face quadrant. The eyebrow region is then obtained by clipping the triangle defined by the eye points (the corners and the top of the eye found by one of the eye detectors) out of the eye-eyebrow region. Depending of the colour of the eyebrow (dark or light), the eyebrow region is thresholded. After a unique colour is assigned to each of the objects in the scene, the largest is selected and the rest of the objects are discarded. The 4-connected chain codes [31] are applied to localise the eyebrow contour. At the end, two simplified second-degree curves smooth the obtained contour.
Chain Code Eyebrow implemented in C (extraction) 	To localise the left eyebrow, the upper left face quadrant is first segmented from the facial image using the facial axis found by the module MRP to RFM and the contour of the face found by the module Find Head Contour. The segmented part is thresholded by applying the algorithm of minimum variance clustering [14]. The eye-eyebrow region is then located by analysing the horizontal and the vertical image signature [14]. The signatures are filtered using the closing morphological filters. The width of the region is set to the width between the first and the last index of the maximal value of the smoothed vertical signature. The height of the region is set two the width of the smoothed horizontal signature. The similar procedure of thresholding and segmenting is applied once more in order to define the eyebrow region. The contour of the eyebrow is found by applying the contour-following algorithm based on the 4-connected chain codes [31].
Eye NN implemented in C (extraction)	Neural network approach to eye tracking. The method is described in detail in Section 4.1.
Snake Eye, implemented in C (extraction) 	To localise the box enclosing the eye, the same method is used as in Chain Code Eyebrow module. The algorithm applies further the active contour method proposed by Kass et al. [17] with the greedy algorithm for minimising of the snake's energy function proposed by Williams and Shah [40].

images, remained below 3 pixels. Moreover, in most of the images the localisation error of each micro-feature is approximately 0.5 pixels.

The results of the detection performance on the testing set of images are shown in Table 12. From this table one can see that the localisation error for all of the micro-features remained below 4 pixels and that, in most of the images, the error is approximately one pixel. In fact, most of the larger localisation errors were caused by the difference in the "definition" of the eye centre in the case of manual

Table 10
The modules of the ISFER Workbench integrated in the extraction group – Mouth- and other features detectors

Module	Module description
Curve fitting of the Mouth implemented in C (extraction) 	The box enclosing the mouth, found by the MRP to RFM module, is segmented from the facial image and then filtered with a two-dimensional Gaussian low-pass filter. In the binarised image, the lowest highlighted pixel is then selected as the boundary-following starting point. Directly connected pixel to the current pixel, representing a zero crossing of the second derivative function of the mouth image, continues the mouth boundary. The points where the conjunction of the lips ends and changes in disjunction are marked as the mouth corners. A refined estimate of the mouth shape is then obtained by fitting two second-degree parabolas on the upper lip and a second-degree parabola on the lower lip. The second order least square model algorithm is used to find the best relation between the points of the extracted mouth contour and the parameters of each of the parabolas.
Snake Mouth, implemented in C (extraction) 	The box enclosing the mouth, found by the MRP to RFM module, is first linearly filtered and segmented from the facial image. Then the mouth-through line is found as a distinct valley in the vertical section of intensity. The minimum of the line with lowest horizontal integral projection of intensity, representing the centre of the mouth, is found first. A function of the vertical section of intensity through the found minimum is created next. The minimum of this function is found and then the valley is detected by searching in both directions for edge points (zero-crossings in the second derivation of intensity starting from the previously found minimum). The mouth-through line is further defined using an altered area-growing algorithm. The algorithm starts from the centre of the mouth and adds points that are 4-connected to the current point if their intensity is lower than the mean intensity of the previously found valley. The algorithm applies further the active contour method proposed by Kass et al. [17] with the greedy algorithm for minimising of the snake's energy function proposed by Williams and Shah [40]. The snakes start in the shape of ellipse whose horizontal axis is the mouth-through line, elongated on both sides for 25%.
Fuzzy Mouth, implemented in C++ (extraction)	Fuzzy classifier of the mouth expressions. The method is explained in detail in Section 4.2.
Image intensity in Facial Areas, implemented in C (extraction)	This module is still under development. The used algorithm is based on the results of several modules, namely Chain Code Eyebrow, Eye Points NN and Curve fitting of the Mouth. The image intensity in a facial region (features f19, f20, f21, f22, f23 and f24 of the frontal-view face model) is represented as the area of the vertical signature function obtained for that facial region. The image intensity on the vertical axis of the mouth (feature f25 of the frontal-view face model) is obtained as brightness distribution data [20] along that line.
Find Profile Contour, implemented in Java (extraction) 	Wojdel et al. has presented the profile detector [42]. First the *Value* of HSV colour model is calculated and exploited for the thresholding of the input profile image. The tip of the nose is then found as the most right high-lighted part of the binary image. The tip of the chin is found as the first distinct minimum in the vector of summed background pixels from the bottom. To solve the problem of face rotation, the line between the tip of the nose and the tip of the chin is used as the x-axis of the new co-ordinate system. To obtain the profile contour from the binary image, the number of background pixels are simply counted between the right edge of the image and the first foreground pixel. This obtains a vector that represents a sampling of the profile contour curve. To remove the noise from the contour, an average procedure is performed with a three-pixel wide window, which is slid along the vector. The zero crossing of the 1^{st} derivative of the profile function defined extremes. Usually, many extremes are found (depending of the local profile change). The list of extremes is processed in both directions from the global maximum. The decision about particular extreme rejection is made using two consecutive records in the list. This obtains the list of extremes that reflect the most distinct peaks/ valleys in the profile contour (see Table 5).

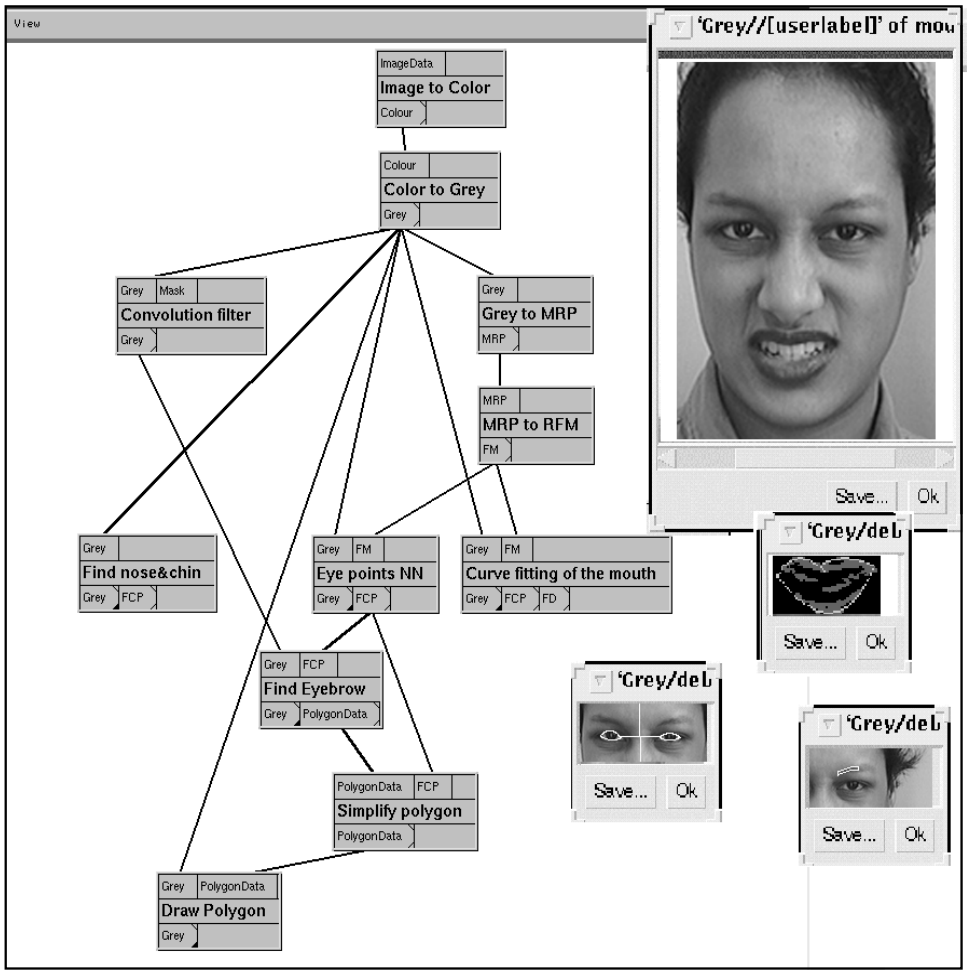


Fig. 9. Screenshot of the ISFER workbench.

estimation and automatic estimation. Manually, the centre of the eye was defined as the centre of the iris while the neural network tends to find the centre as the darkest point of the iris. However, when comparing the performance of different eye-detectors with the performance of our eye-detector similar error distributions are found. The average error of different eye-detectors was measured to be 0.98 pixels for the same resolution of testing images [30].

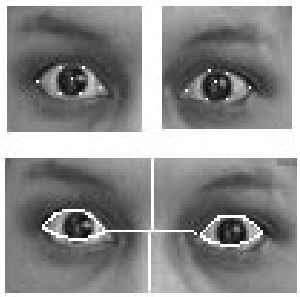


Fig. 10. Curve fitting on the eye micro-features.

Table 11
Distribution of $d(p, p_M)$ for the localised right and left eye micro-features over the training set of 126 images

	0	0.5	1	1.5	2	2.5	3
Right eye							
Left	14	83	22	5	2	0	0
Left-top	5	74	30	9	4	1	3
Right-top	7	71	23	12	6	5	2
Right	16	91	18	0	1	0	0
Right-bot.	8	77	34	6	1	0	0
Left-bot.	6	81	33	2	3	1	0
Centre	8	65	45	8	0	0	0
Left eye							
Left	19	87	18	2	0	0	0
Left-top	4	62	41	12	2	0	5
Right-top	3	68	39	9	2	1	4
Right	22	89	15	0	0	0	0
Right-bot.	11	82	29	4	0	0	0
Left-bot.	9	81	32	2	0	2	0
Centre	9	69	38	10	0	0	0

Table 12
Distribution of $d(p, p_M)$ for the localised right and left eye micro-features over the testing set of 126 images; if $d(p, p_M) = 2.5$ or $d(p, p_M) = 3.5$ the localisation error is counted as 3, 4 pixels, respectively

	0	0.5	1	1.5	2	3	4
Right eye							
Left	9	24	54	37	1	1	0
Left-top	1	17	49	31	24	3	1
Right-top	0	21	43	32	21	3	6
Right	11	19	62	32	0	2	0
Right-bot.	4	17	51	40	12	2	2
Left-bot.	2	20	51	41	9	1	2
Centre	2	42	57	24	1	0	0
Left eye							
Left	11	18	56	39	2	0	0
Left-top	0	12	49	42	15	6	2
Right-top	1	13	53	36	12	7	4
Right	8	26	61	27	4	0	0
Right-bot.	6	21	50	33	13	2	1
Left-bot.	3	19	52	40	4	5	3
Centre	2	47	59	15	3	0	0

4.3. Fuzzy classifier of the mouth expressions

The examination of children's or caricature drawings led us to an interesting conclusion. The mouth expression can be shown using only a single drawing line that still perfectly reflects the intention of the drawer. This led us further to the conclusion that the appropriate representation of the mouth shape may be the information about the average edge intensity and direction in the corners of the mouth. If the edge is on average "going up", mouth would be interpreted as "smiling". If the edge is on average "going down", mouth would be interpreted as "sad". This idea has been implemented by our research team in a form of a fuzzy classifier of mouth expressions [41].

The processing of the module starts with locating the mouth region and the vertical axis of the mouth by applying the MRP to RFM workbench module. Then a fuzzy reasoning for edge detection is performed based on two characteristics of the gradient, namely, the gradient value corresponds with local steepness of the function and the function is locally symmetrical along the gradient direction. The basic idea of fuzzy reasoning for edge detection originates from Law et al. [21]. Still, theirs and our approach differ from each other—our main information is the direction of the gradient rather than its value.

The fuzzy reasoning proceeds as follows. The numerical values representing symmetry and steepness level are first fuzzified into the labels *low*, *medium* or *high* and then passed to the reasoning part of the process. The reasoning part is based on nine rules such as "if the steepness is high and the symmetry level is high then the edge intensity in this point is high". This part results in the labels *low*, *medium* and *high*, which depict the edge intensity in a given point. The information about the direction of the mouth symmetry

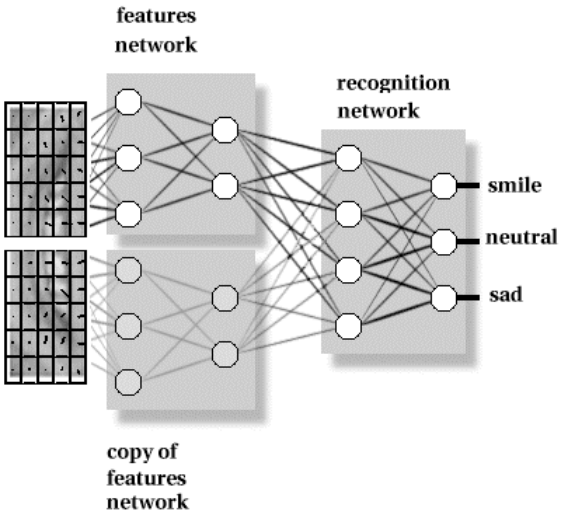


Fig. 11. NN-architecture of the fuzzy classifier.

axis is used to obtain the information about both, the intensity and the direction of the edge in a given point. Combining the intensity and the direction of the edge in a given point results in a vector representation of that point. The obtained vector-field for the whole mouth region is then averaged and 100 average vectors are passed to a $100 \times 6 \times 4 \times 4 \times 3$ backpropagation neural network.

The used network layout (see Fig. 11) reflects the vertical symmetry of the mouth. The implemented architecture contains two $50 \times 3 \times 2$ "features" networks set in parallel (one for each side of the mouth) whose output is passed further to a 4×3 "recognition" network. The output of the network is a singular emotional classification of the shown mouth expression—one of smile, neutral and sad categories.

Both features networks should perform the same task and they can be implemented, therefore, as two copies of the same network. In that case the error is propagated within the single network as well as from the recognition network to both of the features networks. This speeds up the training process and results in better generalisation properties of the network.

To evaluate the method a set of 100 full-face images has been used. The images have been given first to a human observer who classified the images, according to the appearance of the mouth, into one of the three used categories. Then in each experiment, ten images were randomly chosen as the test set. The remaining 90 images were processed first by the fuzzy part of the algorithm and then passed to the network as the training set. In each expirement the network achieved full 100% recognition level for both, the training and the test set of images. The training took in average 60 epochs. Changes in the average error of the network response during the training process are illustrated in Fig. 12. An average response error of the testing set is calculated to be 0.08.

It is not proved yet whether the proposed method is

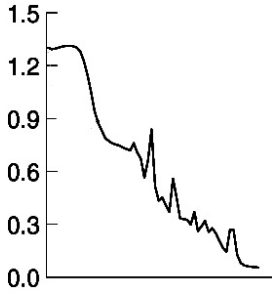


Fig. 12. Average error in training epoch.

sufficiently sensitive. The method uses only some average properties of the image, which do not have to depict subtle differences between various mouth expressions. Still those fine changes in mouth appearance are crucial for a proper (emotional) interpretation of mouth expressions. The method proved quite efficient, however, as a check facility. Overall correctness of the results of other mouth-detectors integrated into the ISFER Workbench can be easily checked on basis of the here presented fuzzy classifier. Using a simple set of rules, the output of the fuzzy classifier can be compared with the properties of the mouth contour localised by another mouth detector. The rules such as "if *smile* then the corners of the mouth are up" extend the fuzzy classifier of mouth expressions and form mouth-detectors checking facility, which has been integrated into the ISFER Workbench as the Fuzzy Mouth module.

5. Dealing with ambiguous facial expression information

In the previous version of ISFER, we used only the frontal-view face model and we tracked the model features by applying the network of workbench modules illustrated in Fig. 9. The accuracy of the resulting face geometry depended, therefore, on the performance of the modules illustrated in Fig. 9. If one of the modules would fail to localise a certain facial feature, we were encountering missing data.

On the other hand, the inference engine of the system (HERCULES) has been developed to reason on a set of exact data about the face [28]. This means that the system would reason on the shown facial expression in a correct way only if all of the necessary facial features have been successfully localised. In the case that one of the modules would fail to localise a certain facial feature, we were substituting the missing data about the currently examined facial expression with the appropriate data from the neutral facial expression. By doing so, accurate information on the examined facial expression was lost.

To enhance the system, we introduced a number of novelties. Two most important are the dual-view face model and the hybrid facial feature detection. The new face model based on a dual facial view resulted in a more realistic and a more reliable representation of the 3D face. The multi-detector operating-mode of the ISFER Workbench ensured maximal reduction of missing data in the output. Namely, for each facial feature a number of detectors based on different algorithms have been built into the ISFER Workbench. Applying several feature-detection approaches, where each one has circumstances under which it performs especially well, reduces the possibility of failed tracking.

Table 13
The priority levels of the workbench extraction modules

Workbench module	Priority
Find Profile Contour	3
Fuzzy Mouth	3
Snake Mouth	2
Curve fitting of the Mouth	3
Snake Eye	2
Eye NN	3
Chain Code Eyebrow	2
Curve fitting of the Eyebrow	3
Find Nose & Chin	2
Image intensity in facial areas (under development)	2
Mouth shapes (under development)	1

After an automatic invoking of each and every detector integrated into the workbench (multi-detector operating-mode), the result of each detector is stored into a separate file. Those files, representing the face geometry of the examined facial expression, form further the input to the Facial Data Evaluator part of ISFER (see Fig. 1).

The information stored in the files is redundant. In the case that none of the detectors of a certain facial feature performed a successful detection, the workbench output files can contain missing data about that facial feature. The workbench output files can also contain highly inaccurate data about some facial feature. As already pointed out while describing our face model, the way of dealing with ambiguous facial expression information depends on a twofold. Namely, the suitability of describing a particular AU with the particular facial view model (given in Table 7) and the overall success ratio of the given detector. We assigned a certain priority to each particular facial feature detector based on both, the information given in Table 7 and the testing results obtained while evaluating the given detector. Based on Table 7, a highest priority has been assigned to the detectors of profile, mouth, eyes and eyebrows. Based on the evaluation results, different priorities were assigned to different detectors of the same facial feature. The facial feature detectors and their priority levels are given in Table 13.

Dealing with ambiguous facial expression information is, in fact, the process of checking, reducing and adjusting the set of files that form the output of the multi-detector operating of the ISFER Workbench. Checking the workbench output is flagging the localised facial features with the labels *good*, *missing*, and *highly inaccurate*. The process of checking the workbench output is described in Section

5.1. The reduction and adjustment of the workbench output is based on the priority levels of the facial feature detectors and on the labels assigned to the localised features in the checking stage of the Facial Data Evaluator. This process is explained in Section 5.2. The result of dealing with redundant, missing and highly inaccurate facial data is unambiguous face geometry. The features defined by our face model can be extracted directly from the obtained face geometry. They form further the input to the ISFER inference engine HERCULES.

5.1. Checking the facial data

Dealing with missing and highly inaccurate data is based on two kinds of knowledge, namely, the knowledge about the neutral facial expression and the knowledge about the facial proportions. When the system is used as an observing tool of a particular person, the pre-processing step of ISFER consists of acquiring a dual-view of the person's neutral facial expression. To ensure correct extraction of the facial features from someone's neutral facial expression, it is highly recommended that the results of automatic feature detection are visually inspected and if necessary, that the choice of facial feature detectors is further manually made. This information is further used to deal with missing and highly inaccurate facial data, encountered in the examined facial expression.

Facial proportions are the facial characteristic distances such as the distance between the eyes and the distance between the centres of the nostrils. The knowledge about someone's facial proportions can be acquired from a full-face image of someone's neutral facial expression. Considering the fact that a 3D face is captured on a 2D image, the facial proportions are not retained if the head is rotated. However, the camera setting defined for ISFER (see Section 1.3), ensures that no rigid head motions can be encountered. Therefore, the position of the head (the size and the orientation) and the facial proportions remain the same in the examined facial expression as in the neutral facial expression in both facial views.

The set of output files resulting from the multi-detector operating of the ISFER Workbench, is evaluated first in terms of missing data. If a single point represents a facial feature, the file containing that feature is labelled as *missing*. In the case of the pair features (eyes and eyebrows), only if a single point represents both facial features the file is labelled as *missing*. If only one of the features is detected as a single point then the file is labelled as *missing one*.

The workbench output files that have not been labelled as *missing* are evaluated further in terms of highly inaccurate data. The evaluation process consists of the following steps.

1. To conclude that the profile contour is badly detected (as a result of for example badly performed thresholding of the profile image), the tip of the nose and the top of the forehead should deviate for at least ten pixels from these points localised in the neutral facial expression. A slight deviation can be also the result of facial muscles contraction (e.g. backwards pull of the ears that pulls the border between the hair and the forehead towards the top of the head). In that case, the file containing the tracked profile contour should not be labelled as *highly inaccurate*.
2. To conclude that the eyes are badly detected one of the following two requirements should be fulfilled. First, the points representing the inner corners of the eyes are immovable points considering the camera setting. If the position of B and B1 (see Fig. 2) deviates for at least five pixels from the neutral expression position of B and B1, one or both eyes will be flagged as badly localised. A slight deviation in the position of the inner corners of the eyes uncovers inaccurate- but no highly inaccurate detection. Although the narrowing and the widening of the eyes can be unilateral, it is almost always bilateral [8]. So, the proportion of one eye comparing to the other should be the same in the examined expression as in the neutral expression. If this is not the case, one or both eyes will be flagged as badly localised. If both eyes are flagged as badly tracked, the file containing the tracked eyes will be labelled as *highly inaccurate*. If only one eye is flagged as badly localised, the file will be labelled as *highly inaccurate one*. This procedure is applied to each file containing the result of an eye detector.
3. In the case of the eyebrows, the important fact is that no muscle contraction can elongate or de-elongate the eyebrow [8]. This and the camera setting, ensure that the area size of each eyebrow remains the same in each examined frontal-view of the observed person. If the size of the eyebrow area deviates for at least ten pixels from the size of that area measured in the neutral facial expression, the eyebrow will be flagged as badly localised. If both eyebrows are flagged so, the file containing this information will be labelled as *highly inaccurate*. If only one eyebrow is flagged as badly localised, the file will be labelled as *highly inaccurate one*. This procedure is applied to each file containing the result of an eyebrow detector.
4. The points representing the centres of the nostrils are immovable points considering the camera setting. So, irrespectively of the performance of the module Find Nose & Chin, the correct location of H and H1 (see Fig. 2) can be always extracted from the neutral facial expression. Still, if the nostrils are not localised correctly by the module Find Nose & Chin, the probability that the tip of the chin is also badly detected is high. If the tracked location of H and H1 deviates for more than five pixels from the neutral expression position of H and H1, the file containing the output of the module Find Nose & Chin will be labelled as *highly inaccurate*.
5. Checking the accuracy of a mouth detector is a pretty difficult task considering the diversity of the possible

mouth movements. The mouth can be elongated or de-elongated, wide open or tightened, puckered or sucked in, laughing or crying. A good way to check if the mouth has been tracked correctly is to compare the tracked mouth shapes in two subsequent frames of a facial image sequence. If the extracted shapes differ a lot, it would be concluded that the mouth has been badly tracked in the currently examined frame. ISFER does not deal, however, with image sequences; it deals with still images. The check that we are performing consists of two steps. First, the opening of the lips (distance KL, Fig. 2) calculated from the localised mouth contour is compared to the distance between the lips calculated from the profile contour (distance P6P8, Fig. 7). If the compared distances deviate for more than five pixels, the file containing the mouth contour will be labelled as *highly inaccurate*. If the mouth contour passes this test, it is checked further with the mouth-detector checking facility implemented as the Fuzzy Mouth module (see Section 4.2). If the mouth contour doesn't pass this second test, the file containing it will be labelled as *highly inaccurate*. This procedure is applied to each file containing the result of a mouth detector. The evaluation explained here, consider only inaccuracies in the vertical stretching of the mouth (and that only in the case that the file containing the profile contour is not labelled as *highly inaccurate*) and inaccuracies in the position of the mouth corners. At the moment, we are not able to take into account inaccuracies in the horizontal stretching of the mouth. This forms a shortcoming of the Facial Data Evaluator.

6. Evaluating the accuracy of the module Image Intensity In Facial Areas and the module Mouth Shapes is not implemented yet, as well as the modules themselves are still under development. However, this does not form a greater shortcoming of the system considering the fact that the results of these modules are not necessary for the processing of the system (see Table 7). Still, in order to facilitate a full-scale reasoning about both facial views, these modules will be integrated into a next version of the system.

At this point, the files that have not been labelled as *missing* or *highly inaccurate* are labelled as *good*.

5.2. Reduction/adjustment of the facial data

After the workbench output files that are labelled as *missing* are discarded, the reduction and the adjustment of the workbench output proceeds as follows.

1. The file containing the profile contour is not discarded even if labelled as *highly inaccurate*. The motivation for doing so is the overall performance of the detection scheme. The algorithm has been tested on 112 profile images representing seven basic emotional expressions shown twice by eight different persons. The images were taken at a resolution of 240×290 pixels and a colour depth of 24 bits. Using Adobe Photoshop and a mouse device, the profile characteristic points (Fig. 7) were manually pointed by a human observer in all 112 images. The performance of the workbench module Find Profile Contour is evaluated by calculating the block distance (maximal difference in x and y direction) between the estimated and the manually located profile characteristic points in each testing image. The performance of the algorithm is shown in Table 14. The localisation error for all profile characteristic points remained below 5 pixels and in most images the error was approximately 2 pixels. Most of the errors were caused by the difference in "definition" of the profile characteristic points in the case of manual and automatic estimation. Manually, the points were defined as the extremes of the profile contour while the automatic scheme tends to find the extremes of the curvature of the profile contour. Anyway, the file containing the output of the module Find Profile Contour will rarely (if ever) be labelled as *highly inaccurate* considering the overall performance of the algorithm with an average localisation error of 2 pixels.

Table 14
Distribution of $d(p, p_M)$ for the localised profile characteristic points (PCPs) over the testing set of 112 images; the localisation error is rounded to a higher integer value

	0	1	2	3	4	5	6
PCP							
P1	8	14	49	27	11	3	0
P2	4	10	43	36	12	7	0
P3	5	9	48	39	9	2	0
P4	12	19	51	29	1	0	0
P5	5	23	47	34	3	0	0
P6	7	20	50	29	2	4	0
P7	3	8	45	40	10	6	0
P8	7	21	49	29	3	3	0
P9	2	10	40	37	15	8	0
P10	11	17	50	28	5	1	0

2. Each workbench output file, which contains the results of an eye detector and has been labelled as *highly inaccurate*, is discarded. If there is no eye-detector file left, the missing data is substituted with the eyes localised in the neutral facial expression. Otherwise, the non-discarded result of the eye detector with a highest priority (see Table 13) will be used in system's further processing. The results of other eye detectors will be discarded in that case. Still, if the eye-detector file with a highest priority is labelled as *missing one* or *highly inaccurate one*, the result of an eye detector with a lower priority will be used to substitute the data about the badly localised eye. If there is no detector with a lower priority, i.e. all are discarded or are labelled as *highly inaccurate one* for the relevant eye, the successfully localised eye will be used to substitute the data about the badly localised eye.

3. In the case of the eyebrows the processing is the same as in the case of the eyes.
4. If the file containing the result of the workbench module Find Nose & Chin is labelled as *highly inaccurate*, the nostrils will be set to the nostrils localised in the neutral facial expression. In that case, the reasoning about the movement of the chin (AU25, AU26 and AU27) will be based on the profile-view model, even if the file containing the profile contour is labelled as *highly inaccurate*. The motivation of doing this is the overall performance of the module Find Profile Contour (Table 14). If the file containing the result of the Find Nose & Chin module is labelled as *good*, while the file containing the profile contour is labelled as *highly inaccurate*, the reasoning about the movement of the chin will be based on the frontal-view model. In that case the rules of Table 7 for AU25–AU27 are replaced by the appropriate rules of Table 4.
5. Each workbench output file, which contains the result of a mouth detector and has been labelled as *highly inaccurate*, is discarded. If there is no mouth-detector file left, the missing data is substituted with the mouth detected in the neutral facial expression. Otherwise, the non-discarded result of the mouth detector with a highest priority (see Table 13) will be used in system's further processing.

The current processing of the ISFER Facial Data Evaluator has several shortcomings. First, the currently implemented data evaluation process will not discover a mouth contour that greatly extends the horizontal length of the actual mouth. Second, all data labelled as *highly inaccurate* will be discarded and, if no data has been labelled as *good*, the relevant facial feature detected in the neutral facial expression will substitute the missing feature. By doing so, accurate information about the examined facial expression gets lost. Finally, ISFER is not able to deal with minor inaccuracies encountered in the workbench output.

To enhance the system's processing we should implement both, dealing with facial image sequences and fuzzy reasoning on facial image data.

A facial expression evolves from a minimal intensity to a maximal intensity (if no emotional shock interrupts it). This means that the global characteristics of a shown expression do not change drastically during a short time interval between two subsequent frames of a facial image sequence. If the system could deal with image sequences, tracking of each facial feature could be checked in correspondence with the relevant feature tracked in the previously examined frame. Obviously, this will enhance the facial data evaluation process.

By implementing a fuzzy reasoning, a certainty level would be assigned to the obtained results. If the assigned certainty is based on the accuracy of the performed facial feature tracking, the system would be able to reason on facial data having any level of inaccuracy. This as well as the aspect of dealing with facial image sequences is currently under development.

The result of the Facial Data Evaluator is the face geometry, unambiguously defined by reduced and adjusted workbench output files. The features defined by our face model (Tables 2 and 5) can be extracted straightforwardly from the files containing the results of feature detectors. The extraction is performed in the Model Data Acquiring step of the system's processing illustrated in Fig. 1. The model features form further the input to the reasoning mechanism of the system.

6. Facial data analysis

The ISFER inference engine is called Human Emotion Recognition Clips Utilised Expert System (HERCULES). The name remained in use although the original version of HERCULES, which dealt exclusively with manually measured frontal-view facial data [28], has been refined to reason on dual-view facial data and form an integral part of ISFER. HERCULES performs automatic facial expression classification in both, AU categories and emotion categories. Classification of expressions in the AU categories is described in Section 6.1. Classification in the emotion categories is explained in Section 6.2.

6.1. Automatic face action tracking

The existing emotion classifiers [2,10,37] singularly categorise examined facial expressions—in one of anger, fear, happiness, surprise, disgust, sadness, and neutral categories. In other words, they are not capable of performing a classification of non-prototypic expressions (such as blends of emotional expressions). In order to develop a system that can recognise complex non-prototypic facial expressions, the face actions should be recognised in the observed face images.

We achieved an automatic face action recognition in two steps. First we perform automatic extraction of the facial features in the examined facial image by utilising the multi-detector processing of the ISFER Workbench. Then the obtained face geometry is automatically converted into a set of activated AUs by utilising the rules of the ISFER inference engine. These rules are given in Table 7 and in the corresponding Tables 4 and 6.

The rules representing the description of the AUs-codes in terms of our combined-view model have been validated twice. First, we asked three certified FACS coders to produce the facial expressions of separate AU activation, according to the rules given in Table 7 and the corresponding tables. Only the changes described in the tables have been produced, the appearance of other facial features is left unchanged. Dual views are recorded and the acquired 96 images (3×32 expressions of separate AU activation, Table 7) were given for evaluation to other two certified FACS coders. In 100% of the cases, the image representing

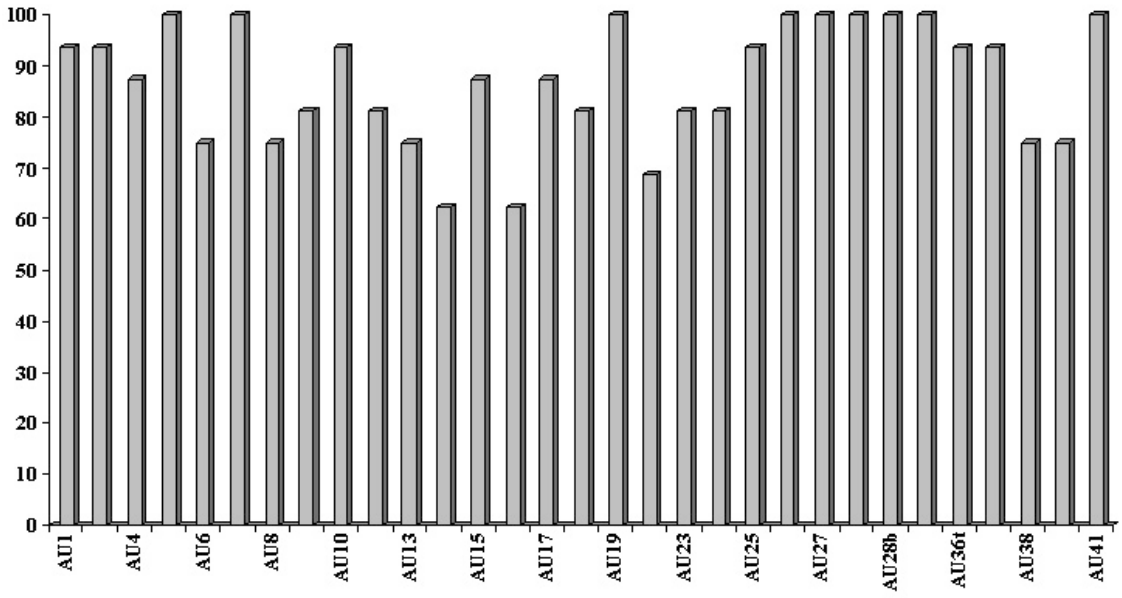


Fig. 13. Face action recognition performance of ISFER when analysed 16 dual views of each of the 31 separately activated AUs.

the activation of a certain AU, produced according to our rules, has been labelled with the same AU-code by the FACS coders. This result was expected, however, considering the fact that all of the rules for representing the AUs-codes in terms of our face model have been generated from the linguistic descriptions given in FACS.

The second validation test of the rules for AU recognition using our face model concerns the automatically performed face action encoding in 496 dual views. The images represent 31 expressions of separate AU activation shown by eight certified FACS coders twice ($2 \times 8 \times 31$). All of the images have been made strictly according to the rules given in Tables 7, 6 and 4. Dual views have been recorded under constant illumination using fixed light sources and none of the subjects had a moustache, a beard or wear glasses. Subjects were of both sexes and ranged in age (22–33) and ethnicity (European, South American and Asian). The average recognition rate was 92% for the upper face AUs and 86% for the lower face AUs (Fig. 13). In only 2% of the images (11 images) the detection failed completely. As expected, a lower recognition rate has been achieved by automatic than by manual facial expression classification in the AU categories. We should point out that, the automatic recognition errors were caused by the localisation errors resulting from the facial feature detectors integrated into the ISFER Workbench and not by some semantic errors of the implemented recognition rules. Improvement of the integrated detectors, that also involves improvement of currently available image processing techniques, will yield system's higher recognition rate of the face actions.

Currently, if the activation of a particular AU is recognised, an intensity level of 100% will be assigned to it. Otherwise, an intensity level of 0% will be assigned to it. We are investigating the methods to make the intensity of an activated AU dependent on the intensity of the related model deformation. We are doing so under the condition that the reasoning of the system remains person-independent.

The production rules that convert the model-based face geometry resulting from the ISFER Facial Data Evaluator into the set of activated AUs, form the first part of the ISFER inference engine (see Fig. 1). The second part of it classifies the shown facial expression in the emotion categories.

6.2. Emotional classification of facial expressions

Classification of the shown facial expression in the emotional categories anger, disgust, fear, happiness, sadness and surprise, is performed by comparing the AUs-coded description of the shown expression to each AUs-coded description of the expression that characterises a particular emotion category. The used AUs-coded descriptions of the expressions characteristic for the six basic emotional categories are given in Table 15. These AUs-coded descriptions have been acquired from the linguistic

Table 15
Description of the six basic emotional expressions in terms of AUs

Expression	Aus-coded description
Happiness	6 + 12 + 16 + (25 or 26)
Sadness	1 + 4 + (6 or 7) + 15 + 17 + (25 or 26)
Anger	4 + 7 + (((23 or 24) with or not 17) or (16 + (25 or 26)) or (10 + 16 + (25 or 26))) with or not 2
Disgust	((10 with or not 17) or (9 with or not 17)) + (25 or 26)
Fear	(1 + 4) + (5 + 7) + 20 + (25 or 26)
Surprise	(1 + 2) + (5 without 7) + 26

Table 16
Distribution of the recognition ratio of 18 emotional expressions made according to the rules of Table 15 and judged by five certified FACS coders

Produced expression	Recognised expression					
	Surprise	Fear	Disgust	Anger	Happiness	Sadness
Surprise	**93**	7	0	0	0	0
Fear	3	**77**	12	0	0	8
Disgust	0	0	**74**	26	0	0
Anger	0	3	15	**82**	0	0
Happiness	2	0	0	0	**98**	0
Sadness	0	6	0	0	0	**94**
						Average: 86.3%

descriptions of the six basic emotional expressions given by Ekman [7,9].

The AUs-coded descriptions of the six basic emotional expressions given in Table 15 represent in fact the production rules of ISFER inference engine (given in a reader-oriented pseudo-code), which are used for automatic classification of the basic emotional expressions. The semantic correctness of the rules has been evaluated in the following way. We asked three certified FACS coders to produce facial expressions according to the rules given in Table 15. Dual views were recorded and the acquired 54 images (3×6 expressions shown by three subjects) were given for evaluation to other five certified FACS coders. Table 16 shows the distribution of the correct recognition ratio and the misrecognition ratio. According to Bassili [1], the correct recognition ratio for the six basic emotional expressions obtained by a trained observer is about 87%. The achieved average of the correct recognition ratio is 86% in the case of utilising our rules to produce emotional expressions. This validates the used rules.

Human faces seldom show "pure" emotional expressions [9]. Most of the time people express "blends" of emotional expressions. A basic emotional expression expressed in a lower intensity than 100% or some combination of the six basic emotional expressions is indicated as a blended emotional expression [9]. In order to deal with non-prototypic emotional facial expressions, we set a hypothesis – *the subsets of AUs-coded description of a basic emotional expression should be classified as the very same emotional expression.* The hypothesis resulted in a set of 43 rules for recognition of blended emotional expressions. The rules are given in Table 17 in a reader-oriented pseudo-code. These rules represent the complete set of the production rules of ISFER inference engine. The rules for recognition of "pure" emotional expressions, given in Table 15, represent the combinations of the rules given in Table 17 and therefore do not exist as such in HERCULES.

The semantic correctness of the rules given in Table 17 has been evaluated as follows. Recordings of three certified FACS coders showing 43 relevant combinations of AUs given in Table 17, made a set of 129 dual-view images. The images have been given then to other five certified FACS coders for judging. Table 18 shows the distribution of the correct recognition ratio. The achieved correct recognition ratio with an average of 85%, in the case of utilising the rules of Table 17 to produce the judged emotional expressions, validates the used rules.

A description of the shown facial expression in terms of weighted emotion labels concludes the facial expression analysis performed by ISFER (Fig. 14). The weight of the assigned emotion label is calculated according to assumption that each AU, forming the AUs-coded description of a particular "pure" emotional expression (Table 15), has the same influence on the intensity of that emotional expression. Let us explain this issue using an example. The facial expression illustrated in Fig. 9 will be classified as the

Table 17
The production rules of ISFER inference engine for emotional classification of facial expressions

AUs	Emotion	AUs	Emotion	AUs	Emotion	AUs	Emotion
1 + 2	Surprise	1	Sadness	23 + 17	Anger	10 + 17	Disgust
2	Anger	4	Anger	23 + 26	Anger	10 + (25/26)	Disgust
6	Happiness	5	Surprise	23	Anger	10	Disgust
1 + 4 + 5 + 7	Fear	7	Anger	24 + 17 + 26	Anger	9 + (25/26)	Disgust
1 + 4 + 5	Fear			24 + 17	Anger	9 + 17	Disgust
1 + 4 + 7	Sadness	27	Surprise	24 + 26	Anger	9	Disgust
1 + 5 + 7	Fear	20 + (25/26)	Fear	24	Anger	12 + (25/26)	Happiness
1 + 4	Sadness	20	Fear	10 + 16 + (25/26)	Anger	12	Happiness
1 + 5	Fear	15 + (25/26)	Sadness	10 + 17 + (25/26)	Disgust	16 + (25/26)	Anger
1 + 7	Sadness	15	Sadness	9 + 17 + (25/26)	Disgust	17	Sadness
5 + 7	Fear	23 + 17 + 26	Anger	12 + 16 + (25/26)	Happiness	26	Surprise

Table 18
Distribution of the correct recognition ratio of 129 emotional expressions made according to the rules of Table 17 and judged by five certified FACS coders

Produced expression	Classified as	Produced expression	Classified as	Produced expression	Classified as	Produced expression	Classified as
1 + 2	95% surprise	1	98% sadness	23 + 17	72% anger	10 + 17	75% disgust
2	77% anger	4	100% anger	23 + 26	49% anger	10 + (25/26)	67% disgust
6	98% happiness	5	79% surprise	23	88% anger	10	82% disgust
1 + 4 + 5 + 7	90% fear	7	92% anger	24 + 17 + 26	85% anger	9 + (25/26)	93% disgust
1 + 4 + 5	93% fear			24 + 17	77% anger	9 + 17	95% disgust
1 + 4 + 7	82% sadness	27	81% surprise	24 + 26	53% anger	9	89% disgust
1 + 5 + 7	96% fear	20 + (25/26)	100% fear	24	92% anger	12 + (25/26)	100% happiness
1 + 4	79% sadness	20	80 % fear	10 + 16 + (25/26)	75% anger	12	100% happiness
1 + 5	81% fear	15 + (25/26)	100% sadness	10 + 17 + (25/26)	79% disgust	16 + (25/26)	68% anger
1 + 7	77% sadness	15	100% sadness	9 + 17 + (25/26)	98% disgust	17	83% sadness
5 + 7	86% fear	23 + 17 + 26	82% anger	12 + 16 + (25/26)	100% happiness	26	70% surprise
							Average: 85.02%

activation of AU9 + AU26. From Table 15 one can see that a facial expression AU10 + AU17 + AU26 or an expression AU9 + AU17 + AU26 will be classified as 100% disgust. So, the facial expression illustrated in Fig. 9 will be classified as 66% disgust, in the case that AU9 and AU26 are 100% activated. Considering the rule for the recognition of AU26 activation (see Tables 6 and 4 and Fig. 15), an intensity level between 0 and 100 can be assigned to the recognised activation according to the extent to which the jaw has been dropped. In the expression given in Fig. 9, the extent to which the jaw has been dropped is measured to be 66.67% of the second-bounding-threshold (i.e. an intensity level of 66.67% has been assigned to the activation of AU26). Therefore, the expression illustrated in Fig. 9 is classified as $0.33 \times 100\% + 0.33 \times 67\% = 55.6\%$ disgust (Fig. 14).

The overall performance of the ISFER automatic emotional classification of facial expressions has been tested on a set of 265 dual-view images. The images represent 129 images used to validate the rules of the ISFER inference engine given in Table 17, 56 images representing 7 "pure" emotional expressions (including neutral expression) and 80 images of various blended emotional expressions (e.g. Figs. 16 and 17) shown by eight certified FACS coders. Dual views have been recorded under constant illumination using fixed light sources and none of the subjects had a moustache, a beard or wear glasses. Subjects were of both sexes and ranged in age (22–33) and ethnicity (European, South American and Asian). First, the images were manually classified according to the rules of Table 17. The performance of the automatic classification is then evaluated by counting the images that have been correctly classified and weighted by the system. In only 2% of the images (6 images) detection failed completely. The average correct recognition ratio was 91% (Table 19).

The encountered errors result from the localisation errors coming from the facial feature detectors integrated into the ISFER Workbench. The localisation errors are causing the errors in the estimation of the shown face actions, and in turn the errors in the emotional classification of the recognised face actions. Improvement of the integrated detectors will yield system's higher recognition rate of the face actions and in turn, a more successful emotional classification of the recognised face actions performed by the system.

One does certainly understand that the rules for emotional classification of facial expression, implemented in ISFER reasoning mechanism are completely based and acquired from Ekman studies [7–9]. Still, one can interpret a shown facial expression differently than Ekman does and therefore differently than the system does.

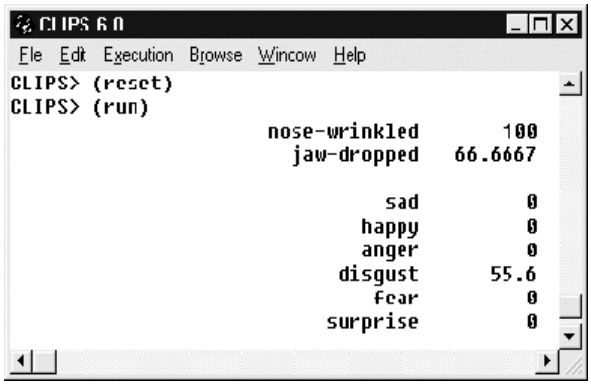


Fig. 14. The screenshot of ISFER output resulting from emotional classification of facial expression illustrated in Fig. 9, priori processed by the ISFER Workbench.

```
(defrule rAU26 "Recognition of AU26"
  (phase recognition-Aus)
  (threshold tAU25 ?tAU25)
  (threshold tAU27 ?tAU27)
  (deviation dCM ?dCM&:(>= ?dCM ?tAU25)
                     &:(< ?dCM ?tAU27))
  ⇒
  (bind ?inte (/ (* ?dCM 100) ?tAU27))
  (assert (activated-AUs 26 ?inte)
          (decription jaw-droppe  ?inte)))
```

Fig. 15. CLIPS-implemented rule for the recognition of AU26 activation based on the frontal-view face model.

Fig. 16. Blend of sad and anger emotional expression.

Fig. 17. Blend of fear and surprise expression.

In spite of the fact that during the last twenty years the theory of Paul Ekman was the most commonly used theory on emotions, correctness and universality of it has been criticised. While Izard [16], Ekman [9], and other psychologists of that school state that the facial appearances of the six basic emotional expressions are indeed universal, another psychological circle doubts even the correctness of the definition of the term "emotion". They argue that the term "facial expression of emotion" used by Ekman is merely a stressing of the verbal communication and has nothing to do with an actual emotional state.

As the goal of our research does not form a solution for this psychological debate but an automatic analysis of a shown facial expression, the best thing to do is to make the system independent of all psychological polemics about emotions. To achieve this and still to retain interpretation facility of the system, a learning facility should be developed. This will allow the user to define his/her own emotion-, or simply interpretation labels. The user can decide then, whether he/she will use the labels predefined by the system or his/her self-defined labels.

7. Conclusion

This paper presents a prototype of the person- and situation independent system for vision-based facial gesture analysis, which utilises a framework for hybrid facial feature detection and an Expert System for face action recognition and emotional classification of facial expressions.

We proposed dual view face model that can recover 32 different face actions (29 AUs) which form an integral part of the human behaviour. The experiments with certified FACS coders indicate that the rules for face action recognition, based on our face model, are valid. The proposed model shapes the face actions globally; it does not consider only local facial regions. The model avoids inaccuracy and manual initialisation of 3D wire-frame models and still it represents a realistic representation of the 3D human face.

A new approach to facial feature detection based on multiple feature detection techniques has been proposed. The modules integrated into the framework for hybrid facial feature detection have been described. By showing the experimental results for two newly developed modules of the ISFER Workbench, we were hoping to give the reader an indication of the overall performance of our framework for hybrid facial feature detection.

The paper has presented a face action recognition strategy based on the proposed face model and multiple feature-detection algorithms applied in parallel. The redundant data, resulting from the multiple-detector operating-mode of the ISFER Workbench, is used to improve the reliability of the system; it is used to solve the problem of missing and erroneous data encountered in the output of the ISFER Workbench. Dealing with this ambiguous facial information

Table 19
Distribution of the correct recognition ratio and the misrecognition ratio of 265 emotional expressions

Expression	Recognised expression						
	Surprise	Fear	Disgust	Anger	Happines	Sadness	B
Surprise	**97**	1	0	0	0	0	2
Fear	0	**84**	0	0	0	9	7
Disgust	0	0	**82**	14	0	0	3
Anger	0	1	12	**84**	0	0	2
Happiness	1	0	0	0	**98**	0	1
Sadness	0	2	0	0	0	**96**	2
B	3	1	0	0	2	1	**93**
							Average: 90.57%

that results from the processing of the ISFER Workbench has been explained in detail. The experimental results indicate that face action recognition can be achieved quite accurately by the system.

In this paper, we also proposed a face action classification strategy that allows singular- as well as multiple classification of facial expression in the six basic emotional categories. By a number of experiments, performed with certified FACS coders, we demonstrated the validity of the rules for the emotional classification of face actions that are implemented in the system. The evaluation of the overall performance of the fully automated system indicates that the facial feature detection, the face action recognition and the face action emotional classification are performed rather accurately by the system.

The system deals with static face action (static image), not with facial motion. Specifically defined camera setting facilitates the system to have no problem with significant head motions. The system does not require use of any special markers or make-up on the user but beard, moustache and eyeglasses are not allowed.

Our ongoing work is focused on a threefold. Modelling the facial motion, i.e. dealing with facial image sequences, will increase the overall performance of the system. Developing a Fuzzy Expert System for face action tracking and face action emotional classification will increase the quality of the system by allowing it to reason about the involved face actions according to the accuracy of the performed facial feature tracking. Designing and developing a learning facility, which will allow the user to define his/her own interpretation categories, will make the system independent of any psychological debate on emotions.

References

[1] J.N. Bassili, Facial motion in the perception of faces and of emotional expression, Journal of Experimental Psychology: Human Perception and Performance 4 (1978) 373–379.

[2] M.J. Black, Y. Yacoob, Recognising Facial Expressions in Image Sequences using Local Parameterised Models of Image Motion, International Journal on Computer Vision 25 (1) (1998) 23–48.

[3] E. Boyle, A.H. Anderson, A. Newlands, The effects of visibility on dialogue and performance in a co-operative problem solving task, Language and Speech 37 (1) (1994) 1–20.

[4] V. Bruce, Recognising Faces, Lawrence Erlbaum, Hove, East Sussex, 1986.

[5] J. Cohn, A.J. Zlochower, J.J. Lien, T. Kanade, Feature-point tracking by optical flow discriminates subtle differences in facial expression, in: Third IEEE International Conference on Automatic Face and Gesture Recognition, 1998, pp. 396–401.

[6] A. van Dam, J.D. Foley, Computer graphics—algorithms and principles, Addison-Wesley, Reading, MA, 1995.

[7] P. Ekman, W.V. Friesen, Unmasking the Face, Prentice Hall, New Jersey, 1975.

[8] P. Ekman, W.V. Friesen, Facial Action Coding System (FACS): Manual, Consulting Psychologists Press, Palo Alto, 1978.

[9] P. Ekman, Emotion in the Human Face, Cambridge University Press, Cambridge, 1982.

[10] I.A. Essa, A.P. Pentland, Coding analysis interpretation and recognition of facial expressions, IEEE Transactions on Pattern Analysis and Machine Intelligence 19 (7) (1997) 757–763.

[12] A.J. Fridlund, P. Ekman, H. Oster, Facial expressions of emotion: Review literature 1970–1983, in: A.W. Siegman, S. Feldstein (Eds.), Nonverbal Behaviour and Communication, Lawrence Erlbaum Associates, Hillsdale, NJ, 1987, pp. 143–224.

[13] A.S. Glassner, Graphics Gems, Academic Press, New York, 1993 (ISBN 0122861663).

[14] R.M. Haralick, L.G. Shapiro, Computer and Robot Vision, Addison-Wesley, Reading, MA, 1992 (ISBN 069434 2015).

[15] L.D. Harmon, M.K. Khan, R. Lash, P.F. Raming, Machine identification of human faces, Pattern Recognition 13 (1981) 97–110.

[16] C.E. Izard, The Face of Emotion, Appleton-Century-Crofts, New York, 1971.

[17] M. Kass, A. Witkin, Terzopoulos, Snake: active contour model, in: First International Conference on Computer Vision, 1987, pp. 259–269.

[18] M. Kato, I. So, Y. Hishinuma, O. Nakamura, T. Minami, Description and synthesis of facial expressions based on isodensity maps, in: T.L. Kunii (Ed.), Visual Computing, Springer, Tokyo, 1991, pp. 39–56.

[19] G.D. Kearney, S. McKenzie, Machine interpretation of emotion: design of a memory-based expert system for interpreting facial expressions in terms of signalled emotions (JANUS), Cognitive Science 17 (4) (1993) 589–622.

[20] H. Kobayashi, F. Hara, Facial interaction between animated 3D face robot and human beings, in: IEEE International Conference on System, Man and Cybernetics, 1997, pp. 3732–3737.

[21] T. Law, H. Itoh, H. Seki, Image filtering, edge detection and edge tracing using fuzzy reasoning, IEEE Transactions on Pattern Analysis and Machine Intelligence 18 (5) (1994) ???.

[22] J.J. Lien, T. Kanade, J.F. Cohn, C.C. Li, Automated facial expression recognition based on FACS action units, in: Third IEEE International Conference on Automatic Face and Gesture Recognition, 1998, pp. 390–395.

[23] B.D. Lucas, T. Kanade, An iterative image registration technique with an application to stereo vision, in: Seventh International Joint Conference on Artificial Intelligence, 1981, pp. 674–680.

[24] A. Mehrabian, Communication without words, Psychology Today 2 (4) (1968) 53–56.

[25] S. Morishima, H. Harashima, Emotion space for analysis and synthesis of facial expression, in: IEEE International Workshop on Robot and Human Communication, 1993, pp. 188–193.

[26] S. Morishima, F. Kawakami, H. Yamada, H. Harashima, A Modelling of Facial Expression and Emotion for Recognition and Synthesis, Symbiosis of Human and Artifact, Elsevier, Amsterdam, 1995 (pp. 251–256).

[27] J. O'Rourke, Computational Geometry in C, Cambridge University Press, Cambridge, 1994.

[28] M. Pantic, L.J.M. Rothkrantz, H. Koppelaar, Automation of nonverbal communication of facial expressions, in: EUROMEDIA 98, SCS International, Ghent, 1998, pp. 86–93.

[30] M.J.T. Reinders, Eye tracking by template matching using an automatic codebook generation scheme, in: Third Annual Conference of ASCI, ASCI, Delft, 1997, pp. 85–91.

[31] G.X. Ritter, J.N. Wilson, Handbook of Computer Vision Algorithms in Image Algebra, CRC Press, Boca Raton, FL, 1996.

[32] L.J.M. Rothkrantz, M. van Schouwen, F. Ververs, J. Vollering, A multimedial workbench for facial expression analysis, in: EUROMEDIA 98, SCS International, Ghent, 1998, pp. 94–101.

[33] J.A. Russell, Is there universal recognition of emotion from facial expression? A review of cross-cultural studies, Psychological Bulletin 115 (1) (1994) 102–141.

[35] G.M. Stephenson, K. Ayling, D.R. Rutter, The role of visual communication in social exchange, Britain Journal of Social Clinical Psychology 15 (1976) 113–120.

[36] D. Terzopoulos, K. Waters, Analysis and synthesis of facial image

sequences using physical and anatomical models, IEEE Transactions on Pattern Analysis and Machine Intelligence 15 (6) (1993) 569–579.
[37] N.M. Thalmann, P. Kalra, M. Escher, Face to virtual face, Proceedings of the IEEE 86 (5) (1998) 870–883.
[38] R.J. van Vark, L.J.M. Rothkrantz, E.J.H. Kerckhoffs, Prototypes of multimedial stress assessment, in: MediaComm 95, SCS International, Ghent, 1995, pp. 108–112.
[39] J.M. Vincent, D.J. Myers, R.A. Hutchinson, Image feature location in multi-resolution images using a hierarchy of multi-layer perceptrons, Neural Networks for Speech, Vision and Natural Language, Chapman & Hall, London, 1992 (pp. 13–29).
[40] D.J. Williams, M. Shah, A fast algorithm for active contours and curvature estimation, Computer Vision and Image Processing: Image Understanding 55 (1) (1992) 14–26.
[41] J.C. Wojdel, L.J.M. Rothkrantz, Mixed fuzzy-system and artificial neural network approach to the automated recognition of mouth expressions, in: Eighth International Conference on Artificial Neural Networks, 1998, pp. 833–838.
[42] J.C. Wojdel, A. Wojdel, L.J.M. Rothkrantz, Analysis of facial expressions based on silhouettes, in: Fifth Annual Conference of ASCI, ASCI, Delft, 1999.
[43] Y.T. Wu, T. Kanade, J.F. Cohn, C.C. Li, Optical flow estimation using wavelet motion model, in: Sixth IEEE International Conference on Computer Vision, 1998, pp. 992–998.
[44] H. Yamada, Visual information for categorizing facial expressions of emotions, Applied Cognitive Psychology 7 (1993) 257–270.
[45] J. Yang, A. Waibel, A real-time face tracker, in: Workshop on Applications of Computer Vision, 1996, pp. 142–147.
[46] J. Zhao, G. Kearney, Classifying facial emotions by backpropagation neural networks with fuzzy inputs, in: International Conference on Neural Information Processing, vol. 1, 1996, pp. 454–457.