Chapter 18
Automated Analysis of Head Pose, Facial Expression and Affect

Robert Niese, Ayoub Al-Hamadi, and Heiko Neumann

Abstract Automated analysis of facial expressions is a well-investigated research area in the field of computer vision, with impending applications such as human-computer interaction (HCI). The conducted work proposes new methods for the automated evaluation of facial expression in image sequences of color and depth data. In particular, we present the main components of our system, i.e. accurate estimation of the observed person's head pose, followed by facial feature extraction and, third, by classification. Through the application of dimensional affect models, we overcome the use of strict categories, i.e. basic emotions, which are focused on by most state-of-the-art facial expression recognition techniques. This is of importance as in most HCI applications classical basic emotions are only occurring sparsely, and hence are often inadequate to guide the dialog with the user. To resolve this issue we suggest the mapping to the so-called "Circumplex model of affect", which enables us to determine the current affective state of the user, which can then be used in the interaction. Especially, the output of the proposed machine vision-based recognition method gives insight to the observed person's arousal and valence states. In this chapter, we give comprehensive information on the approach and experimental evaluation.

18.1 Introduction

In contemporary human-computer interaction (HCI), machine-based vision increasingly gains pace, whereas, besides gesture control, analysis of faces is an important application area. In that field, not only person identification is focused on, but also the deciphering of non-verbal communication through facial expression, as this can provide feedback about user behavior [17]. In automated camera-based

R. Niese (✉) • A. Al-Hamadi
Institute for Information Technology and Communications (IIKT), University of Magdeburg, Magdeburg, Germany
e-mail: Robert.Niese@ovgu.de; Ayoub.Al-Hamadi@ovgu.de

H. Neumann
Institute for Neural Information Processing, University of Ulm, Ulm, Germany
e-mail: Heiko.Neumann@uni-ulm.de

S. Biundo, A. Wendemuth (eds.), *Companion Technology*, Cognitive Technologies,
DOI 10.1007/978-3-319-43665-4_18

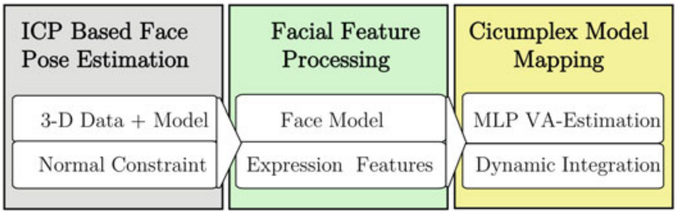

Fig. 18.1 Processing chain of the three main modules

facial expression analysis mostly fixed emotion categories have been utilized in the past, as described by Ekman [6]. However, due to the rare occurrence of classic basic emotions in HCI applications, usually this strategy is of limited use. For that reason, several groups have attempted to map visual and audio-visual utterances, predominantly facial expressions to dimensional emotion-models, like the Circumplex Model of Affect [19], by inferring Valence-Arousal (V-A) parameters. Having these affective user state parameters, the course of the interaction can generally be guided more intuitively, i.e. the machine can provide help to a puzzled user. In the dimensional emotion model, the parameters represent states from negative to positive for valence and calm to aroused for arousal. It has also been shown that the V-A transformation can be disturbed by inaccuracies in the image-based processing [14]. In the presented concept, we focus on this problem through hierarchical analysis. Further, we derive the intensity of a particular expression state, which provides a useful parameter for interaction, i.e. a user with high arousal can be given a different response.

In the following three sections, we present the components of our system as depicted in Fig. 18.1. The first one is used to determine the observed person's head pose, followed by facial feature processing and, third, by classification. In our concept, these are successive modules, which can also be substituted by alternative approaches, i.e. a different pose estimator, feature set, or classification strategy, in order to adapt to a particular application. The evaluation of the presented methods is based on a tilt sensor for the pose, analysis of a 3-D database for facial expressions of emotion, as well as online examples, as shown at the end each section.

18.2 ICP-Based Face Pose Estimation

Automated analysis of image content requires precise knowledge about the arrangement in the captured scene. This does not only include detection and recognition of the interesting objects, but also the determination of their orientation. This general principle also holds for automated analysis of human subjects observed by a camera system, and of course, it is easy to see that the evaluation of a rotated face differs much from a frontal one. Driven by the availability of depth sensors at affordable prices, e.g. Microsoft Kinect, ASUS WAVI Xtion or SoftKinetic, in the recent few years numerous market-feasible applications have been developed for human-machine interaction, mostly for gesture recognition in real time. It has been shown that by using active depth sensors, many problems can be tackled, e.g. illumination changes, strong rotation, occlusion and difficult background. This gives motivation for the presented face pose estimation approach. Fanelli et al. [8] have presented an efficient, but training-intensive depth-based head pose estimation that uses the Discriminative Regression Forest technique. In contrast, our approach achieves a high accuracy and robustness and can handle strong rotations even without excessive training [15]. It is based on an extended variant of the Iterative Closest Point (ICP) registration algorithm, which was originally introduced by Besl and McKay [2]. In the applied ICP approach a user-adapted face model is registered with measured point cloud data. The adaptation is carried out only once in an initialization step. The important processing steps and the used parametrization are given in the following.

18.2.1 Acquisition of 3-D Scene Data

In the first processing step the camera's depth and color data is captured using the software frameworks OpenCV/OpenNI [3]. Under the assumption of a pinhole camera model and a given camera constant, we compute point cloud $\mathbf{W}$ (18.1) of the scene from the depth image [15]. Further, we define a box as the operating volume $\mathbf{V}$ (18.2) that limits the amount of point cloud data and excludes the background (Fig. 18.2). The parametrization depends upon the experimental setup, e.g. we have mounted the camera on top of the monitor.

$$\mathbf{W} = \{\mathbf{p}_1, \dots, \mathbf{p}_n\}, \mathbf{p}_i \in \mathbb{R}^3 \tag{18.1}$$

$$\mathbf{V} = \{\mathbf{p}_{min}, \mathbf{p}_{max}\}, \tag{18.2}$$

with $\mathbf{p}_{min} = (-0.5, -0.5, 0.5)$, $\mathbf{p}_{max} = (0.5, 0.5, 1.0)$ in meters.

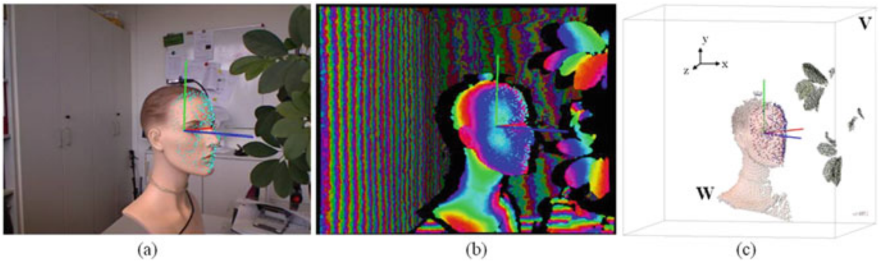

Fig. 18.2 Captured scene. (**a**) Image and pose encoded in a coordinate system (RGB $\simeq$ XYZ) in the centroid of the head, (**b**) depth map, (**c**) point cloud **W** and volume **V**

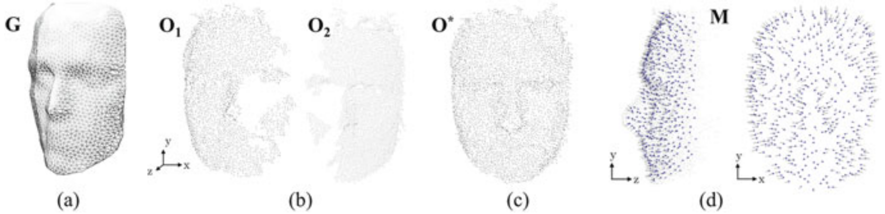

Fig. 18.3 Creation of ICP model. (**a**) Generic model **G**, (**b**) Point clouds $\mathbf{O}_1$ and $\mathbf{O}_2$ of *left* and *right* sides, (**c**) fused point cloud $\mathbf{O}^*$, (**d**) ICP fitting model **M** containing $n = 500$ vertices $\mathbf{a}_j$ and normals $\mathbf{b}_j$

18.2.2 Creation of User-Specific ICP Fitting Model

In this work, we apply a generic geometrical face model **G** (18.3) for the creation of user-specific ICP fitting models, which present the basis for pose estimation. There are several techniques for the creation of 3-D face models; the easiest way is a face scanner [10]. During processing, model **G** represents a smoothed coarse geometric average of ten evaluated subjects [15] (Fig. 18.3a). Thus, model **G** represents a general face shape and consists of a set of vertices $\mathbf{a}_i$, normal vectors $\mathbf{b}_i$ and triangle indices w_j. That enables the model to serve for rough pose estimation of up to ±40° rotation angles when dealing with unknown faces.

$$\mathbf{G} = (\{\mathbf{a}_1, \ldots, \mathbf{a}_n\}, \{\mathbf{b}_1, \ldots, \mathbf{b}_n\}, \{w_1, \ldots, w_m\}),\ \mathbf{a}_j, \mathbf{b}_j \in \mathbb{R}^3, w_k \in \mathbb{N} \qquad (18.3)$$

However, in order to achieve accurate results, especially for large rotation angles, it is beneficial to adapt the model shape and size as close as possible to the actual face. It is to be noted that transient face shape changes that may occur due to facial expression can be neglected at this point, as it has been shown in experiments that these changes do not have relevant influence on the pose estimation with the presented approach [15].

For the creation of the person-specific accurate ICP model, several point clouds $\mathbf{O}_i$ of the respective person are combined from different views, ideally from the left and right sides (±25° rotation) (Fig. 18.3b). For the determination of the points $\mathbf{O}_i$ the generic model **G** is approximated to the captured point cloud **W** (18.1) by using the ICP algorithm as presented in this chapter. Next, all measuring points are used that have a Euclidean minimum distance to model **G**. The combination of all point clouds $\mathbf{O}_i$ is done by utilizing the measured poses of the generic model. Hence, the different measurements are realigned to a common orientation $\mathbf{O}^*$ in the same coordinate system (Fig. 18.3c). Subsequently, the points are triangulated to a mesh, which provides normal vectors for each vertex.

For the reduction of computational cost, while keeping high accuracy it has been proven suitable to sub-sample the triangle mesh to $n = 500$ vertices, in order to create the ICP fitting model **M** (18.4) [15]. The model consists of a set of vertices

$\mathbf{a}_j$ and normal vectors $\mathbf{b}_j$ (Fig. 18.3d).

$$\mathbf{M} = (\{\mathbf{a}_1, \ldots, \mathbf{a}_n\}, \{\mathbf{b}_1, \ldots, \mathbf{b}_n\}),\ \mathbf{a}_j, \mathbf{b}_j \in \mathbb{R}^3,\ n = 500 \tag{18.4}$$

18.2.3 ICP-Based Pose Estimation Using a Normal Vector Constraint

Estimation of rigid body pose usually refers to the determination of six unknown parameters, i.e. three translations and three rotations, which in the following are referred to as pose vector $\mathbf{t}$.

$$\mathbf{t} = (t_x\ t_y\ t_z\ t_\omega\ t_\phi\ t_\kappa)^{\mathrm{T}},\ \mathbf{t}_i \in \mathbb{R} \tag{18.5}$$

Generally, the determination of the pose from image data is an optimization problem, which is mostly solved iteratively on the basis of an error measure. Differences between the pose estimation methods arise in the definition of the error measure, the kind of utilized model and image features and the type of correspondences. In the case of captured 3-D scenes and unknown correspondences between model and world data, the Iterative Closest Point (ICP) algorithm offers opportunities for a quick and accurate solution. In general, in the ICP approach correspondences are determined between two basically n-dimensional data sets, like point clouds or geometrical descriptions, while reducing a global distance measure and approximating the pose parameters.

Accordingly, the computation of the head pose is carried out by aligning the 3-D model $\mathbf{M}$ (18.4) respectively $\mathbf{G}$ (18.3) with respect to point cloud $\mathbf{W}$ (18.1). The error function $e(\mathbf{t})$ (18.6) represents the quality of the current pose $\mathbf{t}$. The total error results from the sum of all squared distances d_j between the model vertices $\mathbf{a}_j$ and the plane, which contains the spatially closest measuring point $\mathbf{p}_i$ in point cloud $\mathbf{W}$. Further, this plane is oriented orthogonally to the model's normal vector $\mathbf{b}_j$ (Fig. 18.4).

$$e(\mathbf{t}) = \sum_j (d_j(\mathbf{t}))^2 \rightarrow \min,\ d_j(\mathbf{t}) = (\mathbf{a}_j(\mathbf{t}) - \mathbf{p}_i) \cdot \mathbf{b}_j, \tag{18.6}$$

with $\mathbf{t} \in \mathbb{R}^6,\ \mathbf{a}_j, \mathbf{b}_j, \mathbf{p}_i, \in \mathbb{R}^3,\ d_j \in \mathbb{R}$.

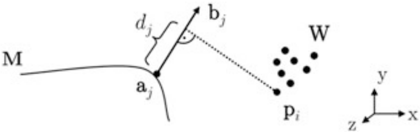

Fig. 18.4 Model fitting principle. Minimization of orthogonal model to point cloud distance d_j. Consider model $\mathbf{M}$ with vertex $\mathbf{a}_j$: searched for is the next point $\mathbf{p}_i$ in point cloud $\mathbf{W}$ lying in a plane perpendicular to $\mathbf{b}_j$

Algorithmically, the ICP method applied is as follows:

- *Initialization or reset of pose vector* $\mathbf{t}^{[0]}$ *if necessary.*
- *Let* $\mathbf{W}$ (18.1) *be a cloud of points* $\mathbf{p}_i$ *and* $\mathbf{M}$ (18.4) *an ICP model with vertices* $\mathbf{a}_j$ *and associated normals* $\mathbf{b}_j$.
- *Repeat for* $k = 1 \ldots k_{max}$ *or until convergence*:
  - *Determine a set of closest correspondences* $\mathbf{S}$
    $\mathbf{S} = \bigcup_{j=1}^{m} (\mathbf{a}_j(\mathbf{t}^k), f_{cp}(\mathbf{W}, \mathbf{a}_j(\mathbf{t}^k)))$
    with f_{cp} returning the closest point $\mathbf{p}_i$ in $\mathbf{W}$ to any point $\mathbf{a}_j$.
  - *Compute the new pose vector* $\mathbf{t}^{[k+1]}$, *which minimizes the fitting error function* $e(\mathbf{t})$ (18.6) *with respect to all pairs* $\mathbf{S}$.

In order to efficiently find the corresponding model points $\mathbf{a}_j$ and measuring points in $\mathbf{W}$ we use function f_{cp}, which applies a kd-search tree [1]. When determining the correspondence, parameter d_{max} defines the maximum allowed distance between model and target points. In this way, a robust out-of-plane rotation is ensured also for large angles, because all target points that are farther away will have no influence on the computation. In particular, we use the empirical threshold $d_{max} = 10$ mm (Fig. 18.5).

The optimization of pose vector $\mathbf{t}$ in error function $e(\mathbf{t})$ (18.6) is carried out iteratively on the basis of least-squares minimization. The elementary matrices for model rotation contain sine and cosine functions, which we need to linearize in order to solve the system of equations for the minimization. This is accomplished using Taylor series approximation. Then, the model vertex coordinates are differentiated with respect to the components of the pose vector. The derivatives $\partial \mathbf{a}_j / \partial \mathbf{t}$ are computed analytically, which can be done easily in the case of translations and rotations. Thus, for each 3-D model point $\mathbf{a}_j$ we can form three equations (18.7),

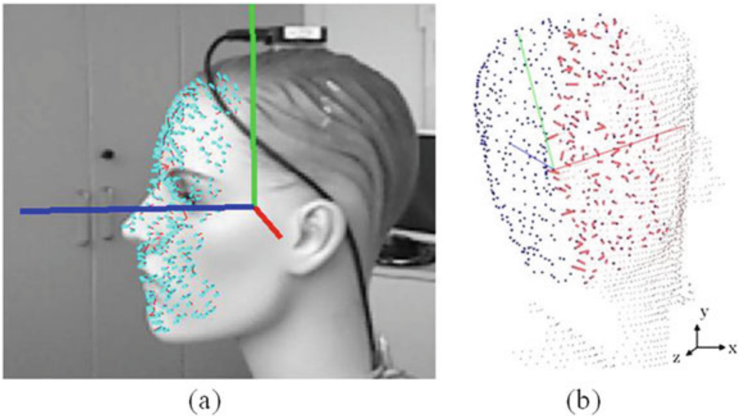

Fig. 18.5 Correspondence search. The *red lines* show associated model and measurement points. (**a**) Image with pose (*XYZ*-axes in RGB) and model in *light blue* plus correspondences in *red*; (**b**) model points of the turned-away face half exceed the maximum distance d_{max} and are no longer associated

which leads to a highly overdetermined system of equations. That itself leads to tolerance against noise and robustness of the computed pose vector.

$$\mathbf{a}_j(\mathbf{t}) + \partial\mathbf{a}_j/\partial\mathbf{t} \cdot \Delta\mathbf{t} = \mathbf{p}_i \tag{18.7}$$

The ICP approach stops if error function $e(\mathbf{t})$ goes below a threshold, or if a given number of iterations has been reached. As shown in comprehensive tests, the computed pose vector **t** accurately represents the actual orientation of the face. For the initialization and reset of the pose vector, it is assumed that the face is located in the upper half of the point cloud, which is generally the case if the camera is aligned properly. With respect to the x- and z-coordinates, we use the centroid of the point cloud, while the rotation angles are set to zero.

Also, in the beginning, and after reset, we apply $n = 20$ iterations, which leads to convergence in most cases. This can be observed through a small value of error function $e(\mathbf{t})$. After initialization, $n = 5$ iterations are applied, or less, if the error goes below threshold $f_{err} = 20$. Reset of the pose is triggered by the error function exceeding the threshold $f_{err} = 40$. That is a hint of misalignment, which can happen if the face is fully occluded. Further, reset is carried out if the number of corresponding model and target points is too low for a reliable computation, i.e. $n_{corr} < 250$. This can happen if no face is in the measurement volume.

Using these error measures, quick and robust pose estimation processing is assured. Alternatively, if it is available, one can also utilize the grayscale image corresponding to the depth map, e.g. for Haar-like feature-based face detection using Adaboost [21]. This way, one can set the initial XY-translation of the 3-D model w.r.t. the image face position, or detect if several or no faces are in the image.

18.2.4 Evaluation of Head Pose Estimation

As initially stated, our pose estimation procedure is essential for automated face analysis. Thus, in order to get a qualitative statement, we have made an evaluation on the basis of the exact tilt sensor 3DM-GX3 of the company MicroStrain, which provides ground truth rotation parameters at high accuracy (Fig. 18.6).

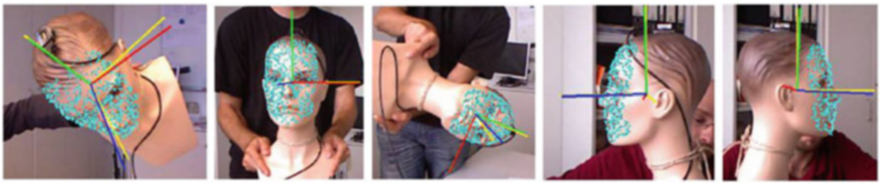

Fig. 18.6 Pose validation using a dummy head model with tilt sensor. The pose is displayed as coordinate system in RGB, the ground truth in *yellow* and the 3-D ICP model in *light blue*

Table 18.1 Evaluation: rotation and tilt sensor displacement in degrees

Rotation	Maximum absolute values in degrees		Ground truth displacement in degrees		
axis	Estimation	Tilt sensor (ground truth)	Mean μ	Std. deviation	Maximum
rx	51	50	2	2	6
ry	108	110	3	3	11
rz	115	116	2	2	4

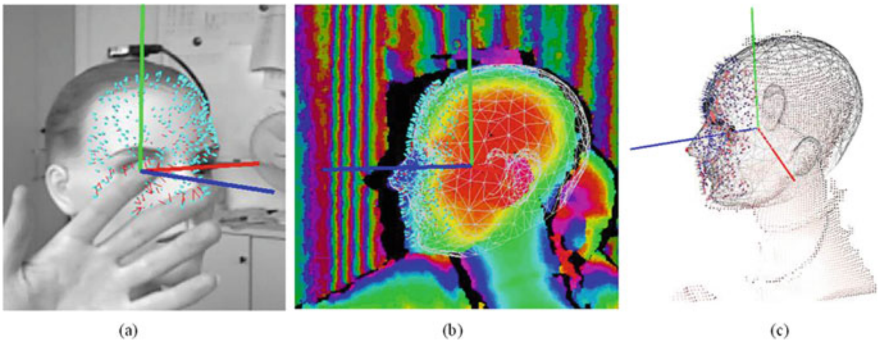

Fig. 18.7 (**a**) Occlusion tolerance, (**b**) application of an additional head model for ongoing facial feature extraction and analysis, (**c**) 3-D point cloud with head model and the determined pose

The estimated pose accuracy has been determined as the ground truth deviation in a series of 10,000 measured sample frames, including rotations of the three axes. Table 18.1 shows the resulting relevant results of the analysis. For the measured data, the tilt sensor has returned a maximum absolute rotation of {*rx ry rz*}={50° 110° 116°}. The first two columns show the computed maximum absolute rotations, the next two the mean pose sensor displacement with corresponding standard deviations, all in degrees. The last column provides the measured maximum ground truth deviations, which have occurred at strong rotations, i.e. for yaw of more than 90°, only. For all rotation angles, the mean deviation is less than 3°.

Robust handling of occlusion is a further concern of the presented work. Due to the characteristics of the measured 3-D data, pose estimation of partially occluded heads is still possible using temporal coherence (Fig. 18.7a). That means the model is fitted step by step up to large rotation angles. The processing speed is high and reaches the maximum possible frame rate (30 Hz) of the used camera's USB port at VGA resolution on an Intel Core i7 PC.

18.2.5 Summary of Pose Estimation

The presented 3-D data-based pose estimation procedure has been shown to perform robustly and accurately in a wide range of head poses. This is achieved through the use of a model normal constraint in conjunction with an ICP algorithm. In particular, the method constitutes a solid basis for the application of 3-D head models for further analysis (Fig. 18.7b/c). Also, in experiments with real persons, we have evaluated that there is only a minimal effect on face pose accuracy related to facial expression, which is due to the fact that the pose model covers the whole of the face. This makes it an ideal basis for subsequent face analysis, which is presented in the following Sect. 18.3.

18.3 Facial Feature Processing

In image-based analysis of faces, one can discern at least three categories of commonly used features, such as holistic vs. geometric analytic, two- vs. three-dimensional, i.e. image- or volume-based and temporal vs. static features [17]. In this categorization, almost all face recognition approaches apply holistic, 2-D image-based, static features, whereas in facial expression analysis there are analytic geometric region-based and dynamic features commonly used. In our work, apart from the 3-D depth image-based pose estimation, which is done in the first step, we apply a combination of 2-D image and 3-D model data for the processing of geometric features.

18.3.1 Face Model

In the presented method, facial feature processing and evaluation are based on a geometric 3-D face model, which utilizes the Facegen Photofit routine [7]. This is a morphable model, which is adapted to a frontal face image using facial landmarks. These are quickly and reliably found using the IntraFace detector by Xiong et al. [23] in conjunction with gradient data and the active contour model algorithm of Cootes [5] (Fig. 18.8a/b).

In order to set the appropriate size of the Facegen-based model, we apply scaling in X- and Y-direction by utilizing point cloud data derived from the depth image and the ICP algorithm of Sect. 18.2 with scaling as free model parameter (Fig. 18.8c/d).

For the conducted face processing we use a rigid 3-D surface mesh description of the adapted Facegen model, which is denoted by **S** (18.8).

$$\mathbf{S} = (\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}, \{w_1, \ldots, w_m\}),\ \mathbf{v}_i \in \mathbb{R}^3,\ w_j \in \mathbb{N}, \tag{18.8}$$

with $\mathbf{v}_i$ as mesh vertices and w_j as triangle indices.

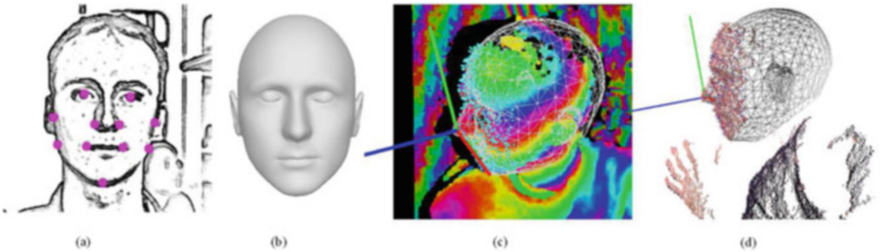

Fig. 18.8 Face model adaptation, (**a**) frontal face gradient image with detected landmark points, (**b**) reconstructed 3-D model **S**, (**c**) depth image encoded in cyclic rainbow colors with scaled model at pose, (**d**) point cloud with scaled model

18.3.2 Facial Expression-Related Features

In the presented concept, geometric features represent the facial expression at frame t through a set of distances and angles. These parameters result from an evaluation of relevant facial feature points.

18.3.2.1 Feature Points

Evaluation of feature points is a general technique in face and facial expression analysis. The Facial Animation Parameter (FAP) System [16], which was developed in the context of the MPEG-4 standard, has inspired the selection of facial points in our method. In the FAP 88 feature points were defined for the simulation of facial expression. In experiments, we found that a subset of eight relevant points suffices for the recognition of facial expression (Fig. 18.9). For this purpose we use point set $\mathbf{P}_f$ (18.9). The model-based computation of feature points requires the detection of the corresponding image points beforehand.

$$\mathbf{P}_f = \{\mathbf{p}_{le}, \mathbf{p}_{re}, \mathbf{p}_{leb}, \mathbf{p}_{reb}, \mathbf{p}_{lm}, \mathbf{p}_{rm}, \mathbf{p}_{ul}, \mathbf{p}_{ll}\},\ \ \mathbf{p}_i \in \mathbb{R}^3, \tag{18.9}$$

with le/re as left and right eye, leb/reb as eyebrow and mouth points (see Fig. 18.9c).

18.3.2.2 Extraction and Transformation of Image Feature Points

For the detection of facial points in the image, we apply a Haar-like feature-based Adaboost face detector in the first step in order to find the face and constrain the facial feature search space [21]. Then we apply the IntraFace detector [23] for feature point finding. To solve non-linear least-squares (NLS) functions, which are

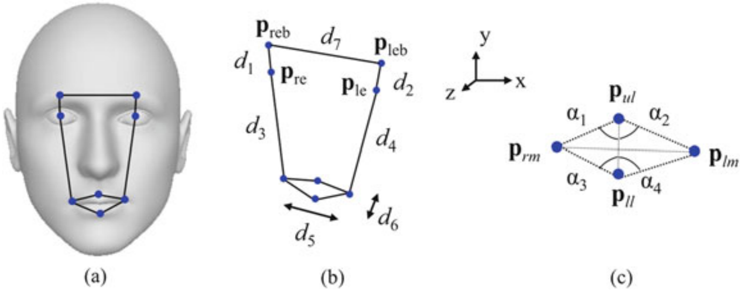

Fig. 18.9 (**a**) Face model **S** with features, (**b**) feature points **p** with distances d_i and (**c**) angles α_j, in the mouth region

taken for feature point detection, the IntraFace method uses the fast and accurate Supervised Descent Method (SDM) approach. This way facial expression relevant points are detected in the input image in a reliable and fast manner, until out-of-plane rotations up to 25°. In the following, these points are referred to as set $\mathbf{I}_f$ (18.10).

$$\mathbf{I}_f = \{\mathbf{i}_{le}, \mathbf{i}_{re}, \mathbf{i}_{leb}, \mathbf{i}_{reb}, \mathbf{i}_{lm}, \mathbf{i}_{rm}, \mathbf{i}_{ul}, \mathbf{i}_{ll}\},\ \mathbf{i}_i \in \mathbb{R}^2 \tag{18.10}$$

The 3-D transformation of image feature points requires knowledge about the underlying camera system and on the other side about the depth of the captured scene. In the case of the applied Kinect camera system, depth information is available. However, as the depth map occasionally contains artifacts, such as holes, a different and more general solution is preferred here, which can be applied to an arbitrary camera system. We infer depth information by measuring the distance d from the camera in the scene at the respective pixel raster coordinate in 3-D to the face model using a raycasting algorithm [9]. The model is oriented in the current estimated pose (Sect. 18.2). These transformation steps require information about the camera model **K**, and thus about intrinsic and extrinsic camera parameters, which are determined through calibration [12]. In this way we can easily define the transformation function k (18.11) that converts 3-D scene points **w** to image points **i** and the other way around as k^{-1} (18.12). Using this transformation the 3-D feature point set **P** (18.9) is determined.

$$\mathbf{i} = k(\mathbf{w}, \mathbf{K}), \tag{18.11}$$

with $\mathbf{i} \in \mathbb{R}^2$, $\mathbf{w} \in \mathbb{R}^3$, camera model **K**.

$$\mathbf{w} = k^{-1}(\mathbf{i}, d, \mathbf{K}), \tag{18.12}$$

with $\mathbf{i} \in \mathbb{R}^2$, $\mathbf{w} \in \mathbb{R}^3$, depth parameter $d \in \mathbb{R}$ and camera model **K**.

18.3.2.3 Feature Definition

The appearance of faces showing expressions differs from the neutral ones in a more or less accentuated way. Strong expressions can, of course, be recognized more easily than weak ones with subtle changes only. Instead of holistic approaches that evaluate the whole of the face at once, we determine expression features from the facial feature points and compare these to the neutral face, which is captured beforehand. It is commonly recognized that the recognition performs better if this neutral information is given [20]. In order to make this kind of comparison more practical, there are already strategies available to overcome this sometimes impossible initialization step, by using Point Distribution Models (PDMs), and thus, to also handle unknown faces [20].

The transition from 2-D image to 3-D facial expression features offers clear advantages, in particular, independence from the face pose. This property is utilized in the evaluation of 3-D feature point set $\mathbf{P}_f$ and the inference of the raw feature vector $\mathbf{f}$ (18.13). The feature vector for the neutral facial expression $\mathbf{f}_{neutral}$ is equivalent to $\mathbf{f}$ and kept for each subject for further processing. The vector's values are seven distances d_i (18.14) distributed across the face and four angles α_j (18.15), which contain distinct information about the current mouth shape and the facial expression as a whole (Fig. 18.9). In particular, raising and lowering of the eyebrows is captured through the parameters d_1 and d_2, mouth movements through the distance between mouth corners and eye centers d_3 and d_4. Additionally, the current mouth width, height and eyebrow distance are encoded in d_5, d_6 and d_7.

$$\mathbf{f} = (d_1 \dots \; d_7 \; \alpha_1 \dots \; \alpha_4)^{\mathrm{T}}, \; d_i, \alpha_j \in \mathbb{R}, \; \mathbf{f} \in \mathbb{R}^{11}, \tag{18.13}$$

with the definition of distances d_i as

$$\begin{aligned} d_1 &= ||\mathbf{p}_{reb} - \mathbf{p}_{re}|| & d_2 &= ||\mathbf{p}_{leb} - \mathbf{p}_{le}|| \\ d_3 &= ||\mathbf{p}_{re} - \mathbf{p}_{rm}|| & d_4 &= ||\mathbf{p}_{le} - \mathbf{p}_{lm}|| \\ d_5 &= ||\mathbf{p}_{rm} - \mathbf{p}_{lm}|| & d_6 &= ||\mathbf{p}_{ul} - \mathbf{p}_{ll}|| \\ d_7 &= ||\mathbf{p}_{reb} - \mathbf{p}_{leb}|| \end{aligned} \tag{18.14}$$

and angles α_j as

$$\begin{aligned} \alpha_1 &= \arccos\left(\frac{\mathbf{v_1} \cdot \mathbf{v_2}}{\|\mathbf{v_1}\| \cdot \|\mathbf{v_2}\|}\right) & \alpha_2 &= \arccos\left(\frac{\mathbf{v_2} \cdot \mathbf{v_3}}{\|\mathbf{v_2}\| \cdot \|\mathbf{v_3}\|}\right) \\ \alpha_3 &= \arccos\left(\frac{-\mathbf{v_2} \cdot \mathbf{v_4}}{\|\mathbf{v_2}\| \cdot \|\mathbf{v_4}\|}\right) & \alpha_4 &= \arccos\left(\frac{-\mathbf{v_2} \cdot \mathbf{v_5}}{\|\mathbf{v_2}\| \cdot \|\mathbf{v_5}\|}\right) \end{aligned} \tag{18.15}$$

with

$$\mathbf{v}_1 = \mathbf{p}_{rm} - \mathbf{p}_{ul}, \quad \mathbf{v}_2 = \mathbf{p}_{ll} - \mathbf{p}_{ul}, \quad \mathbf{v}_3 = \mathbf{p}_{lm} - \mathbf{p}_{ul},$$
$$\mathbf{v}_4 = \mathbf{p}_{rm} - \mathbf{p}_{ll}, \quad \mathbf{v}_5 = \mathbf{p}_{lm} - \mathbf{p}_{ll}, \quad \mathbf{v}_i, \mathbf{p}_j \in \mathbb{R}^3$$

18.3.2.4 Feature Normalization

In order to evaluate the facial expression captured at frame t represented by feature vector $\mathbf{f}$ (18.13), we make a comparison with the neutral face, which is provided through $\mathbf{f}_{neutral}$ as explained above. For this purpose we introduce the operator # (18.16) for component-wise division of two feature vectors $\mathbf{a}$ and $\mathbf{b}$.

$$\mathbf{a} \,\#\, \mathbf{b} = (a_1/b_1 \;\, a_2/b_2 \;\, \ldots \;\, a_n/b_n)^{\mathrm{T}} \in \mathbb{R}^n, \; \mathbf{a}, \mathbf{b} \in \mathbb{R}^n \tag{18.16}$$

$$\mathbf{f}_{ratio}(t) = \mathbf{f}(t) \,\#\, \mathbf{f}_{neutral}, \;\; \mathbf{f}_{ratio}, \; \mathbf{f}, \; \mathbf{f}_{neutral} \in \mathbb{R}^{11}, \; t \in \mathbb{Z} \tag{18.17}$$

The ratios of $\mathbf{f}_{ratio}(t)$ (18.16) usually have large deviations between different persons and facial expressions. Thus, we apply feature normalization. For all vector components of $\mathbf{f}_{ratio}(t)$, we have determined the statistical parameters mean and standard deviation as well as the minimum and maximum, $\mathbf{c}_{min}$ and $\mathbf{c}_{max}$ (18.18), across a representative set of example data. Feature vector $\mathbf{f}(t)$ (18.19) is the normalization result, which is computed for the empirical confidence interval of 2σ.

$$\mathbf{c}_{min} = \mu - 2\sigma, \, \mathbf{c}_{min} \in \mathbb{R}^{11} \tag{18.18}$$
$$\mathbf{c}_{max} = \mu + 2\sigma, \, \mathbf{c}_{max} \in \mathbb{R}^{11}$$

with μ and σ as mean and standard deviation for all vector rows.

$$\mathbf{f}(t) = (\mathbf{f}_{ratio} - \mathbf{c}_{min}) \,\#\, (\mathbf{c}_{max} - \mathbf{c}_{min}) = (\mathbf{f}_{ratio} - \mathbf{c}_{min}) \,\#\, 4\sigma, \;\; \mathbf{f} \in \mathbb{R}^{11} \tag{18.19}$$

18.4 Circumplex Model Mapping

In literature, the majority of approaches for facial expression analysis apply discrete categories, mostly emotions, pain, and sleepiness. However, often fixed categories are not optimal, as they can be mixed and ambiguous. For that reason, the approach we apply is influenced by the observation that the affect labels valence and arousal of the Circumplex Model of Affect lead to a state representation that is continuous in principle [19]. Thus, no discrete descriptions of the user state are necessary for classification.

18.4.1 Multi-Layer Perceptron Based Valence-Arousal Estimation

In order to appropriate the circumplex model, in our work we use a technical implementation of the concept from psychology. For this purpose the transformation of the 11-D feature vector to the 2-D model plane is realized by using mapping function f_{map} (18.21) (Fig. 18.10). In our implementation, the circumplex plane is spanned by six polar coordinates P_{C_i} (18.20) of the discrete emotion categories plus neutral (Fig. 18.11) [14]. Basically, this definition reflects the findings of

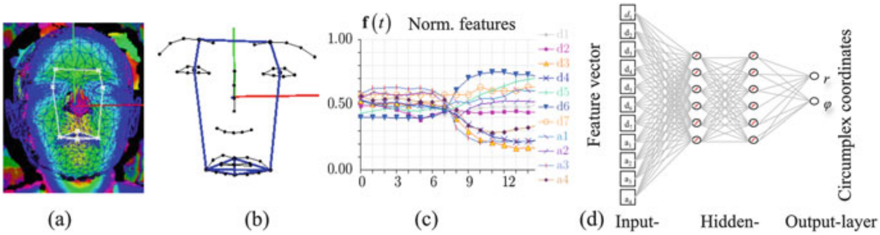

Fig. 18.10 Valence-Arousal (V-A) transformation. (**a**) Depth image with pose and overlaid feature points, (**b**) 3-D features in *blue*, (**c**) feature plot, (**d**) artificial neural network with function f_{map} of the 11-dimensional feature vector to the V-A space

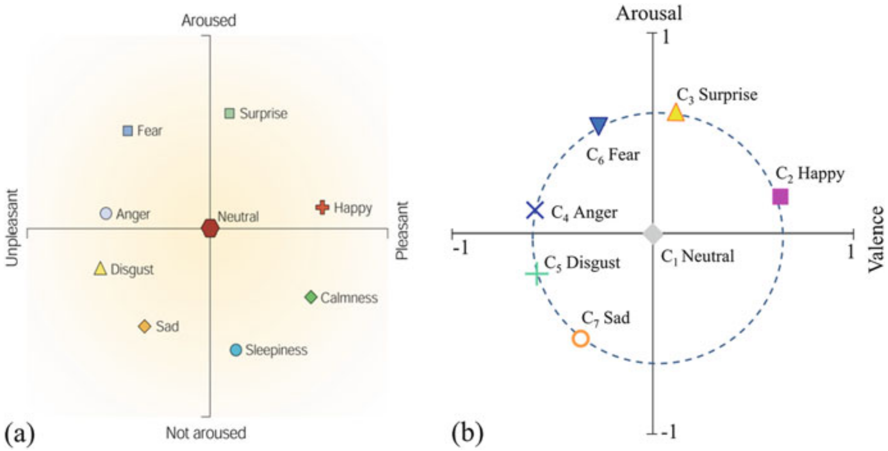

Fig. 18.11 (**a**) Circumplex model of affect as introduced by Russel (Source [4]), (**b**) technical implementation of the model with polar coordinates P_{C_i} (18.20) for the definition of the 2-D model plane

psychologist Russel [19].

$$P_{C_i}(r_{C_i};\varphi_{C_i}) \in \begin{Bmatrix} 0 & l & l & l & l & l & l \\ 0 & 10 & 85 & 170 & 200 & 125 & 240 \end{Bmatrix},\ l = 0.7,\ \varphi_{C_i}\ in\ deg,\ c_i \in 1\ldots 7, \tag{18.20}$$

with the classes $C_i \in$ {Neutral, Joy, Surprise, Anger, Disgust, Fear, Sadness}, whereas radius l has been set empirically to $l = 0.7$.

We apply an artificial neural network (Multi-Layer Perceptron, MLP) [11] with a sigmoid transfer function and a backpropagation training algorithm to realize the transformation f_{map} (18.21) of a feature vector $\mathbf{f}$ at a given frame t. In particular, we apply a network with eleven input and two output neurons plus two hidden layers with six neurons each. Our hypothesis is that we can infer the 11-D to 2-D transformation based on the adapted weights of the neural network, which have been determined through supervised learning. That means, in the training phase each feature vector is assigned to the polar coordinate of its emotion class C_i (18.20), which is derived from the Circumplex Model definition [4]. Accordingly, during classification, each input vector leads to a position in the model plane, which is supposed to be at a place that corresponds to the presented emotion.

$$f_{map} : \mathbf{f}(t) \in \mathbb{R}^{11} \xrightarrow{MLP} \begin{bmatrix} V \\ A \end{bmatrix} \in \mathbb{R}^2, \tag{18.21}$$

with valence V, arousal A.

18.4.2 Dynamic Integration and Determination of Intensity

The estimation of the current affective state can be considered from the viewpoint of inverse problems. In the first stage, the current state is estimated by the evaluation and fusion of optical features using function f_{map} (18.21), leading to an observation in the 2-D V-A space. Now, the goal is to find the underlying unknown state at the current frame t, which is denoted by the variable $z(t)$. From a mathematical point of view, this inverse problem is ill-posed in Hadamard's sense since the reconstruction is potentially sensitive to noise and not guaranteed to be unambiguous. The solution is assumed to be within close distance of the observations, as measured by square norm e_{data}.

$$e_{data} = \|z(t) - f_{map}(\mathbf{f}(t))\|^2,\ z, f_{map} \to \mathbb{R}^2,\ t \in \mathbb{Z} \tag{18.22}$$

Further, the potential solution is constrained by applying operator $P(z)$, in order to achieve a smoothing of the result. The smoothness property is denoted by the first-order derivative of the desired solution, i.e. $P(z) = \dot{z}$,[1] which is scaled by the regularization parameter λ, that works as weighting constant. Taken together, the resulting energy measure is defined as the sum of e_{data} (18.22) and the smoothness constraint $\dot{z}$ defined above,

$$E(z) = \int \|z(t) - f_{map}(\mathbf{f}(t))\|^2 + \lambda \cdot \dot{z}^2(t)dt \rightarrow min. \tag{18.23}$$

For minimization of Eq. (18.23) we apply the Euler-Lagrange equation to solve the partial differential system of equations, which leads to state variable $z(t)$. The intensity level r (18.24) of the current emotion quantity is inferred from the state variable while the user state is traced over a temporal period and integrated over time.

$$z(t) = \begin{pmatrix} r \\ \beta \end{pmatrix}(t), \text{ with } r(t) = \sqrt{a^2 + v^2},\ \beta(t) = \tan^{-1}(v/a), \tag{18.24}$$

with a and v as scalar activations in the cardinal dimensions arousal and valence.

18.4.3 Evaluation

The different modules of the proposed concept have been tested with training and testing data from the BU-4DFE database [24] and exemplary online samples that were taken with a Kinect camera system. A total amount of about 18,000 data samples from seven classes according to (18.20) has been used for training and testing the neural network and the dynamic integration. The pose estimation approach presented in Sect. 18.2 was adapted in order to work with the 3-D data of the BU-database, i.e. to carry out the required 2-D/3-D transformations, image and depth data have been generated through OpenGL rendering [9] with a virtual camera and specified parameters (Fig. 18.12). Analysis has been conducted by applying feature extraction and processing plus V-A mapping to the processed BU-4DFE data (Fig. 18.13).

One motivation of this work is to overcome the use of fixed classification categories, i.e. basic emotions. However, for the evaluation of the V-A mapping, it can be reasonable to apply them. In order to get a qualitative statement about the recognition accuracy, we have analyzed the displacement $\mu(t)$ (18.25) between the calculated angle $\beta(t)$ (18.24) and the given orientation of the underlying class φ_{C_i} in the Circumplex Model's V-A space (Fig. 18.14a).

$$\mu(t) = |\beta(t) - \varphi_{C_i}| \in \mathbb{R}, \text{ see Eqs. (18.20), (18.24)} \tag{18.25}$$

[1] We use here the dot notation to denote temporal derivatives of a function with time as the independent variable.

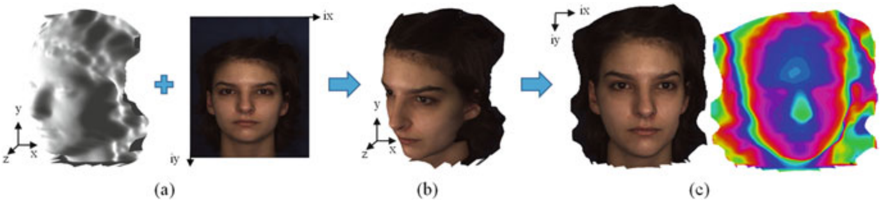

Fig. 18.12 Preprocessing of BU-4DFE data. (**a**) 3-D model and high-resolution texture image, (**b**) textured triangle mesh in 3-D, (**c**) image projection with defined camera as color and depth image

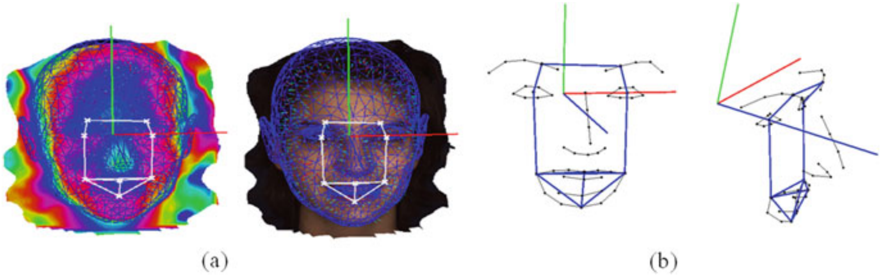

Fig. 18.13 Feature processing of a BU-sample. (**a**) Depth and color image with pose as RGB-coordinate system; further face model **S** (18.8) is shown as *blue triangle* mesh along with extracted features as *white lines*. (**b**) 3-D projection of facial features

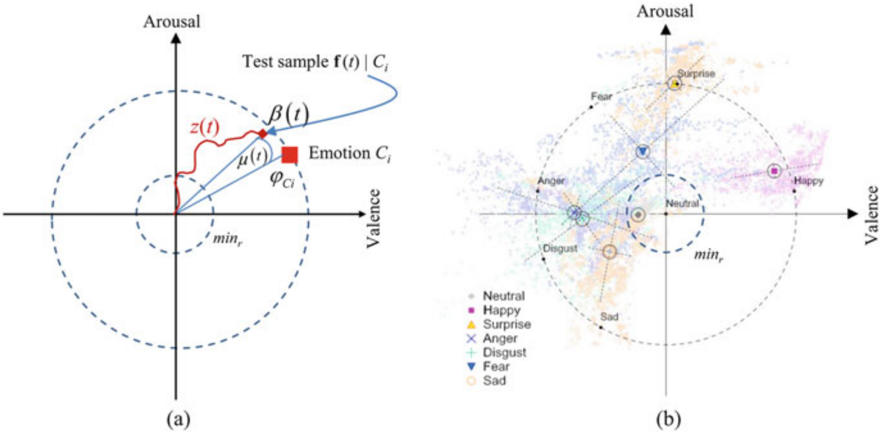

Fig. 18.14 Evaluation. (**a**) The method's accuracy for an exemplary sample $\mathbf{f}(t)$ is determined using angle $\mu(t)$, which reflects the displacement between function $z(t)$ and the given angle φ_{C_i} of the associated class C_i in V-A space. (**b**) Projection of the applied test set from the BU-4DFE database is shown in *light colors* with centroid and principal axes for each class. Data with $r(t) < min_r$ (18.24), i.e. samples with very low expression intensity, are attributed to the neutral class

Table 18.2 Confusion matrix in percent, $t_\mu = 30$

Class	$P(C_1)$	$P(C_2)$	$P(C_3)$	$P(C_4)$	$P(C_5)$	$P(C_6)$	$P(C_7)$
C_1	89.1	0	0.1	0.4	1	0.1	0.3
C_2	3.1	82.2	13.3	0	0	1.4	0
C_3	1.4	0	93.1	0.1	0.1	5.3	0
C_4	0.8	0.5	0	64.5	31.6	1.2	1.4
C_5	6.2	1.6	0.2	37.8	50.4	3.5	0.3
C_6	4.7	5	14.4	7.3	5.9	62.6	0.1
C_7	6.9	0	1.8	6.5	29.5	0.1	53.2

Table 18.3 Confusion matrix in percent, $t_\mu = 60$

Class	$P(C_1)$	$P(C_2)$	$P(C_3)$	$P(C_4)$	$P(C_5)$	$P(C_6)$	$P(C_7)$
C_1	89.1	0	0.1	0.4	1	0.1	0.3
C_2	3.0	89.0	6.6	0	0	1.4	0
C_3	1.4	0	98.4	0.1	0.1	0	0
C_4	0.8	0.5	0	65.7	31.6	0	1.4
C_5	6.2	1.6	0.2	37.8	50.7	3.5	0
C_6	4.7	5.0	1.4	0	5.9	83.0	0
C_7	8.9	0	1.8	6.5	0	0.1	82.8

The neutral class C_1 is treated in a different way, i.e. in the recognition step a sample is considered neutral if $r(t) < min_r$ according to (18.24). The threshold we have determined empirically and set to $min_r = 0.25$. The image material for the neutral class has been taken from the first frames of the database videos. Per definition there is a neutral facial expression at the start of every video. In order to carry out the evaluation we have split the database into training and testing samples in a randomized way, such that all the classes are represented equally and no training sample is taken for testing, which leads to an amount of about 1400 samples per class. In the recognition step, a sample is classified as correct if the angle μ is below threshold t_μ; otherwise it is attributed to the closest adjacent class in the model plane (Fig. 18.14). Tables 18.2 and 18.3 show the resulting confusion matrices for two empirical thresholds t_μ, i.e. $t_\mu = 30°$ and 60°. It becomes obvious how the recognition rate is increasing along with threshold t_μ, because more samples are counted as valid. The confusion among the classes C_6(Fear) and C_3(Surprise) as well as C_7(Sad) and C_5(Disgust) clearly shows this. Further, it can be seen that the highest recognition rates occur for classes with the strongest feature distinction, i.e. Surprise and Happy, whereas confusion occurs for the other classes. The average recognition rates are 70.2 and 79.7 % for $t_\mu = 30$ and $t_\mu = 60$. Basically, these recognition rates are in accordance with category-based state-of-the-art methods for this particular database [22]. Concise inspection of the data shows that even for the human observer, the presented facial expressions cannot always be interpreted correctly. In this case, the continuous description of the user's facially expressed emotional state in the V-A space can provide more opportunities for evaluation, compared to the solely category-based recognition. Figure 18.14b shows the mapping of all used test samples. Also, here the overlapping of samples becomes obvious, in particular for the classes disgust and anger, as well as sad and surprise and fear.

Using a Kinect camera system, we have evaluated the dynamic integration with the temporal smoothness constraint. Exemplary sequences of presented basic emotions are given in Fig. 18.15. Besides extracted features, the temporally smoothed V-A space projections are shown, along with the evolution of function $z(t)$ (18.24), starting from the neutral state in the center. The examples show that the presented expressions become clearly separated, which enables evaluation in terms of valence and arousal of the emotion model.

An example of the smoothing effect of the dynamic temporal constraint is given in Fig. 18.16, which corresponds to first plot in Fig. 18.15a. Also, the underlying feature sequence plus category-based classification is shown. It demonstrates the competitive abilities of the V-A classification approach, which exemplarily shows the detection of a smile. When it comes to evaluation of intensity, the V-A approach

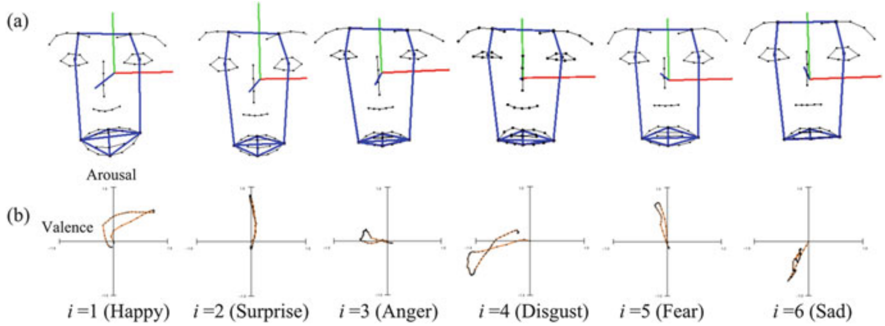

Fig. 18.15 (**a**) 3-D facial expression model with features in *blue*. (**b**) Plot of the projections of presented classic basic emotions in the V-A space using the temporal constraint

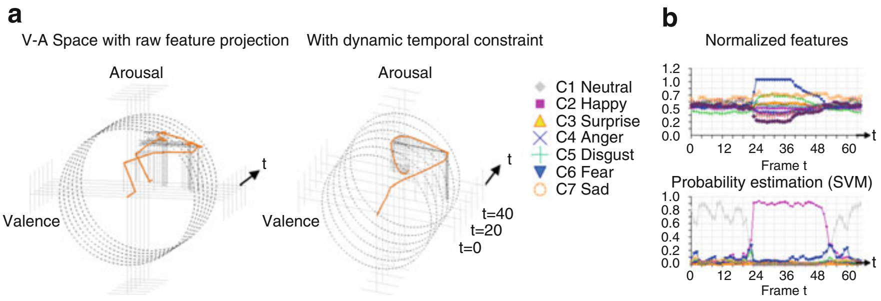

Fig. 18.16 Data processing example with a comparison between affect model and category-based classification. (**a**) Valence-Arousal mapping without and with dynamic temporal constraint, (**b**) corresponding feature vector along with category-based SVM classification

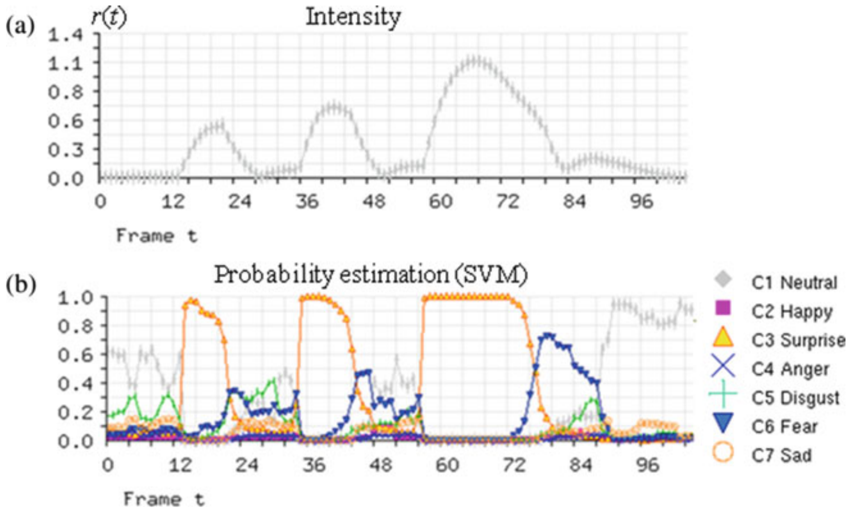

Fig. 18.17 Measuring intensity: Presentation of emotions with different intensity. (**a**) Intensity function $r(t)$ (18.24). (**b**) Note that the respective category-based classification does not provide information about intensity, which shows the advantage of the V-A space-based evaluation

is superior to category-based recognition, as shown in Fig. 18.17. In this example emotions of different expression intensities are presented. Here the category-based classification cannot reflect information about intensity, as the V-A approach does. Thus, it clearly shows the advantage of the applied mapping, which can provide information about intensity.

18.4.4 Summary and Conclusion

In this chapter, we present a concept for facial expression analysis based on image data, which accomplishes the mapping of high-dimensional feature data to the Circumplex model's valence-arousal plane, including dynamic integration and determination of intensity. For feature processing, we apply accurate 3-D depth data-based pose estimation (Sect. 18.2) and feature normalization (Sect. 18.3). As the results show, the presented concept provides more information about the affective state of the user than conventional approaches can deliver when using fixed target classes, like basic emotions, since these rarely occur in HCI applications and also can be ambiguous, e.g. when detecting fear and surprise simultaneously. Even though overlapping may exist after V-A transformation, too, this is not necessarily a bad property, and it is even unavoidable to a certain degree. Overlapping simply results from the fact that some emotions are near to each other in the affect model [19]. As such, it can be useful if the evaluation rather shows a tendency of the user's reaction, i.e. “negtive/positive/aroused or not”, than a particular discrete emotion, which is unlikely to happen. Thus, for an HCI system, this information can be more valuable, such that the presented concept has a potential impact on applicability [14].

In future work, we plan several modifications to the three main parts of our concept. First, in order to facilitate applicability, we want to generalize the adaptation of the user-specific models that are used in the processing. Next we will increase the machine’s perceptive capabilities through the use of advanced sensor technology, like NIR as well as high-speed cameras, and also apply new detection algorithms, like the one of [18], which can robustly and quickly provide a greater number of image features. Further, we also want to acquire and adapt to new application domains. Moreover, at the moment, we neglect the lower right part of the circumplex model plane. This part contains states such as sleepiness and calmness. In future work, we also want to address this quadrant, as it bears potential for vigilance recognition in medical projects, as well as sleep detection in automotive applications [13].

Acknowledgements This work was done within the Transregional Collaborative Research Centre SFB/TRR 62 “*Companion*-Technology for Cognitive Technical Systems” funded by the German Research Foundation (DFG).

References

1. Bentley, J.L.: Multidimensional binary search trees used for associative searching. Commun. ACM **18**(9), 509–517 (1975). http://doi.acm.org/10.1145/361002.361007
2. Besl, P.J., McKay, N.D.: A method for registration of 3-d shapes. IEEE Trans. Pattern Anal. Mach. Intell. **14**(2), 239–256 (1992). doi:10.1109/34.121791. http://doi.ieeecomputersociety.org/10.1109/34.121791
3. Bradski, G.: The OpenCV library. Dr. Dobb’s J. Softw. Tools **25**(11), 120–126 (2000)
4. Calder, A.J., Lawrence, A.D., Young, A.W.: Neuropsychology of fear and loathing. Nat. Rev. Neurosci. **2**(5), 352–363 (2001). doi:10.1038/35072584. http://dx.doi.org/10.1038/35072584
5. Cootes, T.F., Taylor, C.J., Cooper, D.H., Graham, J.: Active shape models - their training and application. Comput. Vis. Image Underst. **61**, 38–59 (1995)
6. Ekman, P.: Strong evidence for universals in facial expressions: a reply to Russell’s mistaken critique. Psychol. Bull. **115**, 268–287 (1994)
7. Facegen modeller. http://facegen.com/modeller.htm (June 2017)
8. Fanelli, G., Gall, J., Gool, L.J.V.: Real time head pose estimation with random regression forests. In: The 24th IEEE Conference on Computer Vision and Pattern Recognition, pp. 617–624 (2011)
9. Foley, J.D., van Dam, A., Feiner, S., Hughes, J.: Computer Graphics: Principles and Practice, 3rd edn. Addison-Wesley, Boston (2013)
10. GFMesstechnik GmbH Germany, T.: Facescan 3d. http://www.gfm3d.com (2015)
11. Haykin, S.: Neural Networks: A Comprehensive Foundation, 3rd edn. Prentice-Hall, Upper Saddle River, NJ (2008)
12. McGlone, C., Mikhail, E., Bethe, J.: Manual of Photogrammetry, 5th edn. ASPRS, ISBN: 1-57083-071-1 (2004)

13. Niese, R., Al-Hamadi, A., Panning, A., Brammen, D.G., Ebmeyer, U., Michaelis, B.: Towards pain recognition in post-operative phases using 3d-based features from video and support vector machines. Int. J. Digit. Content Technol. Appl. **3**(4), 21–33 (2009)
14. Niese, R., Al-Hamadi, A., Heuer, M., Michaelis, B., Matuszewski, B.: Machine vision based recognition of emotions using the circumplex model of affect. In: International Conference on Multimedia Technology (ICMT), pp. 6424–6427 (2011)
15. Niese, R., Werner, P., Al-Hamadi, A.: Accurate, fast and robust realtime face pose estimation using kinect camera. In: IEEE SMC International Conference, pp. 487–490 (2013)
16. Pandzic, I.S., Forchheimer, R.: MPEG-4 Facial Animation: The Standard, Implementation and Applications, 1st edn. Wiley, New York (2002). ISBN: 0-470-84465-5
17. Pantic, M., Pentland, A., Nijholt, A., Huang, T.S.: Human computing and machine understanding of human behavior: a survey. In: Artificial Intelligence for Human Computing, ICMI, pp. 47–71 (2007)
18. Ren, S., Cao, X., Wei, Y., Sun, J.: Face alignment at 3000 FPS via regressing local binary features. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, 23–28 June 2014, pp. 1685–1692 (2014)
19. Russell, J.A.: A circumplex model of affect. J. Pers. Soc. Psychol. **39**, 1161–1178 (1980)
20. Saeed, A., Al-Hamadi, A., Niese, R., Elzobi, M.: Frame-based facial expression recognition using geometrical features. Hindawi Adv. Hum.-Comput. Interaction (2014). doi:10.1155/2014/408953
21. Viola, P.A., Jones, M.J.: Robust real-time face detection. Int. J. Comput. Vis. **57**(2), 137–154 (2004)
22. Wang, J., Yin, L., Wei, X., Sun, Y.: 3d facial expression recognition based on primitive surface feature distribution. In: IEEE International Conference on Computer Vision and Pattern Recognition, CVPR06, pp. 1399–1406 (2006)
23. Xiong, X., De la Torre, F.: Supervised descent method and its applications to face alignment. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition, pp. 532–539 (2013)
24. Yin, L., Chen, X., Sun, Y., Worm, T., Reale, M.: A high-resolution 3d dynamic facial expression database. In: IEEE International Conference on Automatic Face and Gesture Recognition (FG'08), pp. 1–6 (2008)