Fully Automatic Recognition of the Temporal Phases of Facial Actions

Michel F. Valstar, *Member, IEEE*, and Maja Pantic, *Senior Member, IEEE*

***Abstract*—Past work on automatic analysis of facial expressions has focused mostly on detecting prototypic expressions of basic emotions like happiness and anger. The method proposed here enables the detection of a much larger range of facial behavior by recognizing facial muscle actions [action units (AUs)] that compound expressions. AUs are agnostic, leaving the inference about conveyed intent to higher order decision making (e.g., emotion recognition). The proposed fully automatic method not only allows the recognition of 22 AUs but also explicitly models their temporal characteristics (i.e., sequences of temporal segments: neutral, onset, apex, and offset). To do so, it uses a facial point detector based on Gabor-feature-based boosted classifiers to automatically localize 20 facial fiducial points. These points are tracked through a sequence of images using a method called particle filtering with factorized likelihoods. To encode AUs and their temporal activation models based on the tracking data, it applies a combination of GentleBoost, support vector machines, and hidden Markov models. We attain an average AU recognition rate of 95.3% when tested on a benchmark set of deliberately displayed facial expressions and 72% when tested on spontaneous expressions.**

***Index Terms*—Facial expression analysis, GentleBoost, particle filtering, spatiotemporal facial behavior analysis, support vector machine (SVM).**

I. Introduction

FACIAL EXPRESSIONS synchronize the dialogue by means of brow raising and nodding, clarify the content and intent of what is said by means of lip reading and emblems like a wink, signal comprehension, or disagreement, and convey messages about cognitive, psychological, and affective states [1], [2]. Therefore, attaining machine understanding of facial behavior would be highly beneficial for fields as diverse as computing technology, medicine, and security in applications like ambient interfaces, empathetic tutoring, interactive gaming, research on pain and depression, health support appliances, monitoring of stress and fatigue, and deception detection. Because of this practical importance [3], [4] and the theoretical interest of cognitive and medical scientists [5], [6], machine analysis of facial expressions attracted the interest of many researchers in computer vision and AI.

Two main streams in the current research on automatic analysis of facial expressions consider facial affect (emotion) detection and facial muscle action detection [7]–[10]. These streams stem directly from the two major approaches to facial expression measurement in psychological research [11]: message and sign judgment. The aim of the former is to infer what underlies a displayed facial expression, such as affect or personality, while the aim of the latter is to describe the "surface" of the shown behavior, such as facial movement or facial component shape. Thus, a frown can be judged as "anger" in a message-judgment approach and as a facial movement that lowers and pulls the eyebrows closer together in a sign-judgment approach. While message judgment is all about interpretation, sign judgment is agnostic, independent from any interpretation attempt, leaving the inference about the conveyed message to higher order decision making. Most facial expression analyzers developed so far adhere to the message judgment stream and attempt to recognize a small set of prototypic emotional facial expressions such as the six basic emotions proposed by Ekman [7]–[9], [12]. Even though the automatic recognition of the six basic emotions from face images and image sequences is considered largely solved, reports on novel approaches are published even to date (e.g., [13]–[16]). Exceptions from this overall state of the art in machine analysis of human facial affect include few tentative efforts to detect cognitive and psychological states like interest [17], pain [18], [19], and fatigue [20].

In sign judgment approaches [21], a widely used method for manual labeling of facial actions is the Facial Action Coding System (FACS) [22]. FACS associates facial expression changes with actions of the muscles that produce them. It defines 9 different action units (AUs) in the upper face, 18 in the lower face, and 5 AUs that cannot be classified as belonging to either the upper or the lower face. Additionally, it defines so-called action descriptors, 11 for head position, 9 for eye position, and 14 additional descriptors for miscellaneous actions (for examples, see Fig. 1). AUs are considered to be the smallest visually discernible facial movements. FACS also provides the rules for the recognition of AUs' temporal segments (onset, apex, and offset) in a face video. Using FACS, human coders can manually code nearly any anatomically possible facial expression, decomposing it into the specific AUs and their temporal segments that produced the expression. As AUs are independent of any interpretation, they can be used as the

Manuscript received April 28, 2010; revised March 31, 2011 and July 15, 2011; accepted July 24, 2011. Date of current version December 7, 2011. This work was supported by the European Community's Seventh Framework Program [FP7/20072013] under Grant agreement no. 231287 (SSPNet). The work of Michel Valstar was supported by the Engineering and Physical Sciences Research Council Grant EP/H016988/1: Pain rehabilitation: E/Motion-based automated coaching. The work of Maja Pantic was supported by the European Research Council (ERC) under the ERC Starting Grant agreement no. ERC-2007-StG-203143 (MAHNOB). This paper was recommended by Associate Editor X. Jiang.

M. F. Valstar is with the Department of Computing, Imperial College London, SW7 2AZ London, U.K. (e-mail: Michel.Valstar@imperial.ac.uk).

M. Pantic is with the Department of Computing, Imperial College London, SW7 2AZ London, U.K., and also with the Faculty of Electrical Engineering, Mathematics, and Computer Science, University of Twente, 7500 AE Enschede, The Netherlands (e-mail: m.pantic@imperial.ac.uk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMCB.2011.2163710

Fig. 1. Examples of upper and lower face AUs defined in the FACS.

basis for any higher order decision making process including the recognition of basic emotions [22], cognitive states like (dis)agreement and puzzlement [23], psychological states like pain [24], and social signals like emblems (i.e., culture-specific interactive signals like a wink, coded as left or right AU46), regulators (i.e., conversational mediators like the exchange of a look, coded by AUs for eye position), and illustrators (i.e., cues accompanying speech like raised eyebrows, coded as AU1+AU2) [25]. Hence, AUs are extremely suitable to be used as midlevel parameters in an automatic facial behavior analysis system as they reduce the dimensionality of the problem [26] (thousands of anatomically possible facial expressions [25] can be represented as combinations of 32 AUs).

It is not surprising, therefore, that automatic AU coding attracted the interest of computer vision researchers. Historically, the first attempts to encode AUs in images of faces in an automatic way were reported by Bartlett *et al.* [27], Lien *et al.* [28], and Pantic *et al.* [29]. The focus of the research efforts in the field was first on automatic recognition of AUs in either static face images or face image sequences picturing facial expressions produced on command. Several promising prototype systems were reported that can recognize deliberately produced AUs in either (near) frontal view [30]–[32] or profile view face images [32], [33] (for a survey of the past work on the topic, see [9] and [10]).

One of the main criticisms that these works received from both cognitive and computer scientists is that the methods are not applicable in real-life situations, where subtle changes in facial expression typify the displayed facial behavior rather than the exaggerated AU activations typical of deliberately displayed facial expressions. Hence, the focus of the research in the field started to shift toward automatic AU recognition in spontaneous facial expressions (produced in a reflex-like manner). Just recently, few works have been reported on machine analysis of AUs in spontaneous facial expression data [34]–[37] (for a survey, see [8]). These methods employ probabilistic, statistical, and ensemble learning techniques, which seem to be particularly suitable for automatic AU recognition from face image sequences [8], [35], and are either feature or appearance based.

Automatic recognition of facial expression configuration (in terms of AUs constituting the observed expression) has been the main focus of the research efforts in the field. However, both the configuration and the dynamics of facial expressions (i.e., the timing and the duration of various AUs) are important for the interpretation of human facial behavior. In fact, the body of research in cognitive sciences, which argues that the dynamics of facial expressions are crucial for the interpretation of the observed behavior, is ever growing [2], [38]. Facial expression temporal dynamics are essential for the categorization of complex psychological states like various types of pain and mood [24]. They are also the key parameter in the differentiation between posed and spontaneous facial expressions [1]. In spite of these findings, the vast majority of the past work in the field does not take the dynamics of facial expressions into account when analyzing shown facial behavior. Some of the past work in the field has used aspects of temporal dynamics of facial expression such as the speed of a facial point displacement or the persistence of facial parameters over time. However, this was mainly done either in order to increase the performance of facial expression analyzers (e.g., [39]–[41]) or in order to report on the intensity of (a component of) the shown facial expression (e.g., [41] and [42]), but not in order to analyze explicitly the properties of facial actions' temporal dynamics. The only work reported up to date that addresses the problem of modeling semantic and temporal relationships between AUs forming a facial expression is that by Tong *et al.* [40]. Note, however, that this work does not report on the explicit analysis of temporal segments of AUs (e.g., the duration and the speed of the onset and offset of the actions).

Exceptions from this overall state of the art in the field include three studies on automatic segmentation of AU activation into temporal segments (neutral, onset, apex, and offset) in frontal- [43], [44] and profile-view [33] face videos. The works by Pantic and Patras [33], [44] employ rule-based reasoning and geometry-based features to encode AUs and their temporal segments, while Koelstra and Pantic [43] use appearance-based features and hidden Markov models (HMMs).

Fig. 2 outlines our fully automatic detector of 22 AUs and their temporal activation models (from, in total, 27 upper and lower face AUs defined in FACS [22]). This set of 22 AUs contains all upper and lower face AUs that can be robustly recognized based upon movements of 20 facial characteristic points shown in Fig. 2. Although this set is incomplete, the system can be used to encode all but three AUs necessary for the recognition of basic emotions [22], all AUs necessary for the recognition of pain [24], all but one AUs necessary to detect cluelessness [23], and 2/3 of the AUs involved in speech [22] (see Table I for a detailed list).

The method operates under the assumption that the first frame of an input video sequence shows a nonoccluded

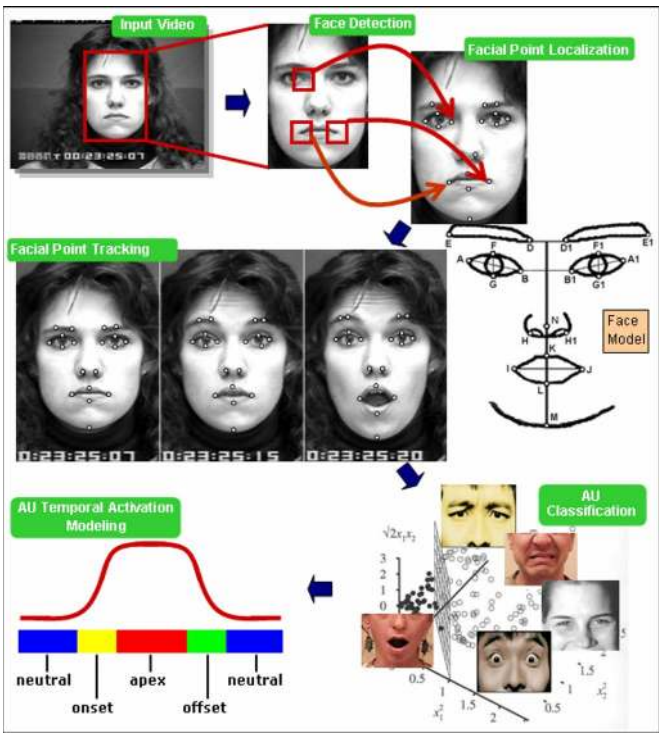

Fig. 2. Outline of the proposed fully automated system for recognition of AUs and their temporal activation models.

TABLE I
AUs DEFINED IN FACS [22], THOSE THAT OUR SYSTEM CAN AUTOMATICALLY ENCODE, AND LISTS OF AUs INVOLVED IN SOME EXPRESSIONS

	AUs
FACS:	upper face: 1, 2, 3, 4, 5, 6, 7, 43 , 45, 46; lower face: 10, 11, 12, 13, 15, 16, 17, 18, 20, 22, 23, 24, 25, 26, 27, 28; other: 9, 21, 31, 37, 38
our system encodes:	1, 2, 4, 5, 6, 7, 9, 10, 12, 13, 15, 16, 18, 20, 22, 24, 25, 26, 27, 43, 45, 46
anger:	4, 5, 7, 10, 17, 22, 23, 24, 25, 26
disgust:	9, 10, 16, 17, 25, 26
fear:	1, 2, 4, 5, 20, 25, 26, 27
happiness:	6, 12, 25
sadness:	1, 4, 6, 11, 15, 17
surprise:	1, 2, 5, 26, 27
pain:	4, 6, 7, 9, 10, 12, 20, 25, 26, 27, 43
cluelessness:	1, 2, 5, 15, 17, 22
speech:	10, 14, 16, 17, 18, 20, 22, 23, 24, 25, 26, 28

expressionless face in near-frontal view. While the method can handle occlusions like facial hair and glasses in general, it cannot handle large amounts of facial hair and/or sunglasses covering one or more facial components completely. Also, while it can handle brief interim occlusions (e.g., by hand), it cannot handle an occluded face in the first frame. After the face region is detected in the first frame, we employ a facial point detector based on Gabor-feature-based boosted classifiers to automatically localize 20 facial fiducial points in the detected face region. To track these points in the rest of the sequence, we exploit a tracking scheme based on particle filtering with factorized likelihoods (PFFL). Using the tracking data, we first detect the presence (i.e., activation) of 22 AUs. We do so by using a combination of GentleBoost ensemble learning and support vector machines (SVMs). For each activated AU, we determine the temporal activation model as a sequence of temporal segments (neutral, onset, apex, and offset). To attain this, we combine GentleBoost, SVMs, and HMMs.

The authors have developed three earlier versions of the AU detector presented in this paper, a 2005 version [45], a 2006 version [46], and a 2007 version [47]. The 2005 version was aimed at automatic recognition of 15 AUs, it was not fully automated, and it did not deal with any temporal information. The 2006 version of the system was aimed at automatic recognition of 15 AUs and their temporal segments (rather than their temporal activation models). Differently from previous versions, the current version is fully automated and aimed at the recognition of 22 AUs and their temporal activation models. The system described in this work is the first that is able to explicitly model the temporal dynamics of AUs in terms of its temporal phases. Also, this work describes extensive tests on databases of posed facial expression data as well as on spontaneous facial expression data. To allow future work to evaluate their methods against the one proposed here, we will make frame-by-frame labeling of the temporal AU segments publicly available (see Section V).

The outline of this paper is as follows. Section II provides an explanation of the employed facial point detector. Section III presents the utilized facial point tracking scheme. Section IV explains the methodology used to detect AUs and their temporal activation models. Section V describes the data sets we used in our validation studies, which are discussed in Section VI. Section VII concludes this paper.

II. FACIAL POINT DETECTION

The first step in any facial information extraction process is face detection, i.e., the identification of all regions in the scene that contain a human face. The second step in facial expression analysis is to extract *geometric features* (facial points and shapes of facial components) and/or *appearance features* (descriptions of the texture of the face such as wrinkles and furrows). The work presented here is a typical example of a geometric-feature-based method. Typical examples of appearance-based methods are those of Bartlett *et al.* [35], [42], [48], who used Gabor filters, or of Anderson and McOwan [14], who used a holistic monochrome spatial-ratio face template, and Jiang *et al.* who used local binary patterns, local phase quantization, and their temporal extensions [49]. Typical examples of hybrid, geometric-, and appearance-feature-based methods are those of Tian *et al.* [31], who used shapes of facial components and transient features like crowfeet wrinkles, or of Zhang and Ji [41], who used 26 facial points and the same transient features as those used by Tian *et al.* [31].

A. Face Detection

Because of its practical importance and relevance to face recognition and, in turn, for security, face detection received a lot of attention. Numerous techniques have been developed [50]–[52]. However, virtually all of them can detect only (near-) upright faces in (near-) frontal view. Most of these methods emphasize statistical learning techniques and use appearance

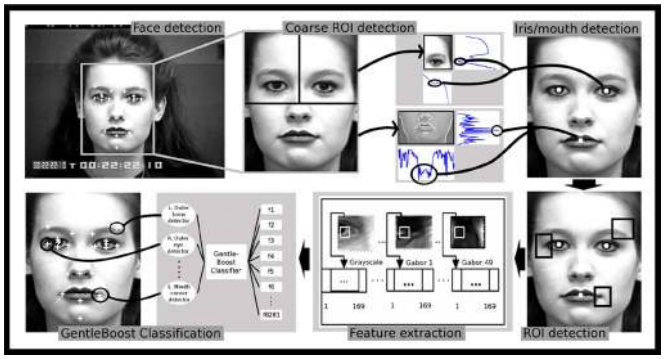

Fig. 3. Outline of the fully automated fiducial facial point detection method.

features, including the real-time Viola–Jones face detector [53], which is arguably the most commonly employed face detector in automatic facial expression analysis.

The Viola–Jones face detector consists of a cascade of classifiers trained by AdaBoost. Each classifier employs integral image filters, which remind of Haar basis functions and can be computed very fast at any location and scale. This is essential to the speed of the detector. For each stage in the cascade, a subset of features is chosen using a feature selection procedure based on AdaBoost. We employed a version of this face detector [54], which was trained on 5000 faces and 8000 nonface images. For images of faces in near-frontal view, it performs very well; for example, when tested on the Cohn–Kanade (CK) database (CK-db) [55], it attained a 100% detection rate [56]. The C++ code of the face detector runs at about 500 Hz on a 3.2-GHz Pentium 4.

B. Characteristic Facial Point Detection

Methods for facial feature point detection can be classified as either *texture-based methods* (modeling local texture around a given facial point) or *shape-based methods* (which regard all facial points as a shape that is learned from a set of labeled faces). A typical texture-based methods is that of Holden and Owens [57], who used log-Gabor filters. Typical texture- and shape-based methods are those of Chen *et al.* [58], who applied AdaBoost to determine facial feature point candidates for each pixel in an input image and used a shape model as a filter to select the most likely position of feature points, and of Cristinacce and Cootes [59], [60], who experimented with various facial point template representations and various search algorithms for finding the best matching shape.

Although these detectors can be used to localize the 20 facial characteristic points illustrated in Fig. 3, none perform the detection with high accuracy. They usually regard the localization of a point to be successful if the distance between the automatically labeled point and the manually labeled point is less than 30% of the true interocular distance D_I (the distance between the eyes, more specifically between the inner eye corners). However, this is an unacceptably large error in the case of facial expression analysis since subtle changes in the facial expression will be missed due to the errors in facial point localization.

We therefore adopt the fiducial facial point detector proposed by Vukadinovic and Pantic [56]. When used to initialize a point tracking algorithm, this method is accurate enough to allow geometric-feature based expression recognition (see the results in Section VI). The outline of the developed fully automated method for the detection of the target 20 facial characteristic points is illustrated in Fig. 3. The method first detects the face and divides the face region into three areas that contain the left eye, the right eye, and the mouth. The locations of these facial components are approximated by analyzing the histograms in the regions. Based on this approximate location, search regions are defined for every point to detect. In these regions of interest (ROIs), a sliding window approach search is performed. At each location of the ROI, Gabor-filter responses are calculated and fed into the GentleBoost-based point detectors. The location with the highest output determines the predicted point location.

Typical results of this algorithm are illustrated in Fig. 4. The point detection algorithm is tolerant to changes in illumination as long as they remain locally constant. If illumination is uneven in the direct neighborhood of a facial point, the point detector may fail for that point. A compiled version of the point detector is available from the authors' Web pages. The nonoptimized Matlab code of our face point detector runs at 0.03 Hz on a 3.2-GHz Pentium 4.

III. PFFL for Facial Point Tracking

After the fiducial facial points are found in the first frame, we track their positions in the entire image sequence. Standard optical flow techniques [61]–[63] are commonly used for facial point tracking in facial expression analysis (e.g., standard Lucas–Kanade optical flow [64] is used in [28], [31], and [34], and an "inverse compositional" extension to this is used in [65]).

To omit the limitations inherent in optical flow methods, such as the accumulation of error and the sensitivity to noise, occlusion, clutter, and changes in illumination, some researchers used sequential state estimation techniques to track facial points in image sequences. Both Zhang and Ji [41] and Gu and Ji [20] used facial point tracking based on a Kalman filtering scheme. The derivation of the Kalman filter is based on a state-space model governed by two assumptions [66]: 1) linearity of the model and 2) Gaussianity of both the dynamic noise in the process equation and the measurement noise in the measurement equation. Under these assumptions, the derivation of the Kalman filter leads to an algorithm that propagates the mean vector and covariance matrix of the state estimation error in an iterative manner and is optimal in the Bayesian setting. To deal with the state estimation in nonlinear dynamical systems, the extended Kalman filter has been proposed, which is derived through the linearization of the state-space model. However, many of the state estimation problems, including human facial expression analysis, are nonlinear and non-Gaussian. To overcome the limitations of the classical Kalman filter and its extended form in general, particle filters have been proposed. For a detailed overview of the various facets of particle filters, see [67].

Fig. 4. Typical first-effort results of the proposed facial-point detector for samples from (left to right): CK-db, the MMI database (posed expressions), two images of the MMI database (spontaneous expressions), the triad data set, and two images from a cell phone camera.

The tracking scheme that we adopt is based on particle filtering. The main idea behind particle filtering is to maintain a set of solutions that are an efficient representation of the conditional probability $p(\alpha|Y)$, where α is the state of a temporal event to be tracked given a set of noisy observations $Y = \{y_1, \ldots, y^-, y\}$ up to the current time instant. This means that the distribution $p(\alpha|Y)$ is represented by a set of pairs $\{s_k, \pi_k\}$ such that, if s_k is chosen with probability equal to π_k, then it is as if s_k was drawn from $p(\alpha|Y)$. By maintaining a set of solutions instead of a single estimate (as is done by Kalman filtering), particle filtering is able to track multimodal conditional probabilities $p(\alpha|Y)$, and it is therefore robust to missing and inaccurate data and particularly attractive for estimation and prediction in nonlinear non-Gaussian systems. In the particle filtering framework, our knowledge about the *a posteriori* probability $p(\alpha|Y)$ is updated in a recursive way.

Several researchers used the condensation algorithm to track facial features in face image sequences (e.g., [68] and [69]). However, the algorithm has three major drawbacks. The first is that a large amount of particles that result from sampling from the proposal density $p(\alpha|Y^-)$ might be wasted because they are propagated into areas with small likelihood. Second, the scheme ignores the fact that, while a particle $s_k = \langle s_{k1}, s_{k2}, \ldots, s_{kN} \rangle$ might have low likelihood, parts of it might be close to the correct solution. Finally, the estimation of the particle weights does not take into account the interdependences between the different parts of α.

The extension to the condensation algorithm that we adopt here for facial point tracking is the PFFL proposed by Patras and Pantic [70]. The PFFL algorithm addresses all of the aforementioned problems inherent in the condensation algorithm by extending the auxiliary particle filtering, which addresses the first drawback of the condensation algorithm [71], to take into account the interdependences between the different parts of the state α.

The PFFL tracking scheme assumes that the state α can be partitioned into substates α_i (which, in our case, correspond to the different facial points) such that $\alpha = \langle \alpha_1, \ldots, \alpha_n \rangle$. At each frame of the input image sequence, we obtain a particle-based representation of $p(\alpha|Y)$ in two stages. First, each partition α_i is propagated and evaluated independently by applying one complete step of the auxiliary particle filtering algorithm. This creates a particle-based representation of $p(\alpha_i|Y)$. In other words, at the first stage of the PFFL tracking scheme, each facial point i is tracked for one frame independently from the other facial points. At the second stage, interdependences between the substates are taken into account by means of a scheme that samples complete particles from the proposal distribution $g(\alpha)$, which is defined as the product of the posteriors of each α_i given the observations, i.e., $g(\alpha) = \prod_i p(\alpha_i|Y)$. Finally, each of the particles produced in this way is reweighted by evaluating the joint probability $p(\alpha|\alpha^-)$ so that the set of particles with their new weights represents the *a posteriori* probability $p(\alpha|Y)$.

The adopted observation model [72] is robust to changes in illumination, and it can deal with large occlusions. This polymorphic aspect is necessary as many areas around facial points change their appearance when a facial action occurs (e.g., the mouth corner in a smile).

IV. Recognition of AUs and Their Temporal Activation Models

Contractions of facial muscles alter the shape and location of the facial components. Some of these changes are observable from the movements of 20 facial points, which we track in the input sequence. To classify the movements of the tracked points in terms of AUs and their temporal activation models, changes in the position of the points over time are first represented as a set of midlevel parameters.

A. Registration and Smoothing

Before the midlevel parameters can be calculated, all rigid head motions in the input sequence must be eliminated. Otherwise, we would not be certain whether the value of a given parameter had changed due to facial muscle contraction or due to rigid head movement. We register each frame of the input image sequence with the first frame using an affine transformation T_1 based on three referential points: the nasal spine point and the inner corners of the eyes (see Fig. 3). We use these points as the referential points because contractions of the facial muscles do not affect these points.

Interperson variations in size and location of the facial points are minimized by applying an affine transformation T_2 to every tracked facial point in each frame. T_2 is obtained by comparing the locations of the referential points of a given subject in the first frame with the corresponding points in a selected expressionless "standard" face (the choice of the subject to be used as this "standard" face does not influence the process). Thus, after tracking any of 20 characteristic facial points in an input sequence containing k frames, we obtain a set of coordinates $\langle p_1, \ldots, p_k \rangle$ corresponding to the locations of the pertinent point p in each of k frames. Then, the registered coordinates p_i^r are obtained as

$$p_i^r(t) = T_2\left(T_1\left(p_i(t)\right)\right). \tag{1}$$

Using this registration technique, four out of six degrees of freedom of head movements can be dealt with, and the remaining two can be handled partially. All three head translation degrees of freedom can be handled completely, as well as all in-plane head rotations (i.e., head roll). Out-of-plane rotations (i.e., head pitch and head yaw) can be dealt with as long as the rotation in these dimensions is smaller than approximately 20°.

The tracked points returned by the PFFL tracker contain random noise occurring due to the probabilistic nature of particle filtering. Therefore, we apply a temporal smoothing filter to arrive at a registered set of points P' that contains less noise

$$p'_i(t) = \frac{1}{2w_s+1} \sum_{t-w_s}^{t+w_s} p^r_i(t) \tag{2}$$

where t denotes the frame number and p' and p^r are elements of the collections P' and P_r, respectively. The window sidelobe size w_s to which we apply the temporal smoothing was chosen after visual inspection of the smoothed tracker's output. For the experiments discussed in this work, $w_s = 1$ has been chosen.

B. Midlevel Parametric Representation

Our midlevel parametric representation is inspired by our earlier work [46], [47]. The most basic features that can be computed from the tracked point information are the positions of the points and the distances between pairs of points. We also compute the angle that the line connecting two points makes with the line $y = 0$ (the horizontal axis).

For each point p'_i, where $i = [1:20]$, the first two features are simply its x and y position. We compute the features f_1 and f_2 for every frame t

$$f_1\left(p'_i(t)\right) = p'_{i,x}(t) \tag{3}$$
$$f_2\left(p'_i(t)\right) = p'_{i,y}(t) \tag{4}$$

where $p'_{i,x}$ and $p'_{i,x}$ are the x and y positions of a point, respectively. For all pairs of points $\{p_i, p_j\}$, $i \neq j$, we compute in each frame two features

$$f_3\left(p'_i(t), p'_j(t)\right) = \left\|p'_i(t) - p'_j(t)\right\| \tag{5}$$

$$f_4\left(p'_i(t), p'_j(t)\right) = \arctan\left(\frac{p'_{i,y}(t) - p'_{j,y}(t)}{p'_{i,x}(t) - p'_{j,x}(t)}\right) \tag{6}$$

where arctan is the modified inverse tangent function that corrects for the quadrant that a point is in (i.e., solves the arctangent problem). Feature f_3 describes the distances between two points p'_i and p'_j, and feature f_4 describes the angle that the line connecting p'_i with p'_j makes with the horizontal axis.

The features $\langle f_1, \ldots, f_4 \rangle$ contain only the information about the positions of the points, the distances between them, and the angles that they make with the horizontal at the current instance in time. No information about the relation of these measurements to their values in a frame displaying a neutral expression is encoded. Neither do they encode any information about the rate of change of the values of these features in consecutive frames (e.g., the velocity of a point). To capture this temporal information, we create a new set of features based on the single-frame-based features described earlier.

First, we compute features that describe how much the feature values have changed, relative to their value at the first neutral frame. We do so using the *difference function* $\kappa(\boldsymbol{x}(t))$

$$\kappa\left(\boldsymbol{x}(t)\right) = x(t) - x(0) \tag{7}$$

where $\boldsymbol{x}$ is any time sequence. Using this definition, we compute the following features:

$$\langle f_5(t) \ldots f_8(t) \rangle = \langle \kappa\left(f_1(t)\right) \ldots \kappa\left(f_4(t)\right) \rangle . \tag{8}$$

To determine the rate of change of the feature values at a given time instance t, we compute their first derivative with respect to time. For discretely sampled data, this becomes

$$\frac{d\left(\boldsymbol{x}(t)\right)}{dt} = v\left(x(t) - x(t-1)\right) \tag{9}$$

where v is the sampling rate of the corresponding recording. We use this definition to compute the features

$$\langle f_9(t) \ldots f_{12}(t) \rangle = \langle d\left(f_1(t)\right)/dt \ldots d\left(f_4(t)\right)/dt \rangle . \tag{10}$$

Finally, we calculate three additional temporal features. Within a certain period w_t, we fit the values of the midlevel features parameters f_j, $j \in [1:4]$, to a second-order polynomial: $f_j(t) = at^2 + bt + c$. In this function, t is the frame number at the center of w_t. In our experiments, the temporal window w_t was seven frames long, which we based on research findings that suggest that temporal changes in neuromuscular facial activity last from 1/4 of a second (a blink) to several minutes (a jaw clench) [22], and a frame rate of 25 Hz of our data. Then, for each d and for each f_j, $j = [1:4]$, we define the following midlevel parameters relating to temporal changes in the value of the midlevel parameters $\langle f_1, \ldots, f_4 \rangle$:

$$f_{10+3*j}(f_j) = a, \quad f_{11+3*j}(f_j) = b, \quad f_{12+3*j}(f_j) = c. \tag{11}$$

In total, this results in a 2520-dimensional feature vector for each frame of our input image sequence.

C. Facial AU Classification

Our approach to AU recognition from input image sequences is based on SVMs. SVMs are very well suited for this task because the high dimensionality of the feature space (representation space) does not affect the training time, which instead depends only on the number of training examples. Furthermore, SVMs generalize well even when few training data are provided. However, note that classification performance decreases when the dimensionality of the feature set is far greater than the number of samples available in the training set [73]. The data sets that we use in this study consist typically of less than 250 image sequences of which 10–20 are positive examples with the remainder being negative examples (see Section V). Given that the dimensionality of the utilized feature set is 2520 (see Section IV-B), overfitting to the training set is rather probable. One way to address this problem is to reduce the

number of features to be used to train the SVM. We do so by means of GentleBoost, which is employed in this stage of the system's processing as a feature selection scheme [74].

An advantage of feature selection by a boosting algorithm is that it tries to optimize the actual classification problem instead of reducing the variability in the data overall, which is done by feature reduction techniques such as PCA. As shown by Littlewort *et al.* [42], when an SVM classifier is trained using the features selected by a boosting algorithm (they used AdaBoost in their study), it outperforms both the SVM and the boosting classifier applied directly to facial expression data.

The implementation of the feature selection has been done as follows. As the weak classifier, we use a linear regression function. For every $d \in D$, where D is the set of 22 AUs that our system can recognize in an input sequence, we apply GentleBoost resulting in a set of selected features G_d. To detect 22 AUs occurring alone or in combination in the current frame of the input sequence (i.e., to classify the current frame into one or more of the $d \in D$), we train a separate SVM to detect the activity for every AU. More specifically, we use G_d to train and test the SVM classifier for the relevant AU (i.e., the relevant $d \in D$). The kernel that we have chosen for the SVM was the radial basis function (RBF) kernel, as this performed best in a pilot study comparing the RBF, polynomial, and linear kernels. For each fold of the validation procedure (see Section VI), the SVM parameters were determined independently of the test data in separate cross-validation loops.

D. Temporal Activation Models of Facial AUs

To encode the temporal segments of the AUs found to be activated in the input image sequence, we proceed as follows. An AU can either be in the following phases: 1) the onset phase, where the muscles are contracting and the appearance of the face changes as the facial action grows stronger; 2) the apex phase, where the facial action is at a peak and there are no more changes in facial appearance due to this particular facial action; 3) the offset phase, where the muscles are relaxing and the face returns to its neutral appearance; or 4) the neutral phase, where there are no signs of activation of this particular facial action. Often, the order of these phases is neutral–onset–apex–offset–neutral, but other combinations such as multiple-apex AUs are also possible. Note that AUs having multiple apices are characteristic for spontaneous facial expressions [75].

As every facial action can be divided into the four temporal segments, we consider the problem to be a four-valued multiclass classification problem. In this paper, we compare two approaches to detect an AU temporal model.

1) mc-SVMs: In the first approach, we employ a one-versus-one strategy to multiclass SVMs (mc-SVMs). For each AU and every pair of temporal segments, we train a separate subclassifier specialized in the discrimination between the two temporal segments. This results in $|C|(|C|-1)/2$ subclassifiers that need to be trained, with $C = \{\text{neutral, onset, apex, offset}\}$ and $|\cdot|$ being the cardinality of a set. For each frame t of an input sequence, every subclassifier returns a prediction of the class $c \in C$, and a majority vote is cast to determine the final output c_t of the mc-SVM for the current frame t. To train the subclassifiers, we apply the following procedure using the same set of midlevel parameters that was used for AU detection (see Section IV-B). For each classifier separating classes $c_i, c_j \in C$, $i \neq j$, we apply GentleBoost, resulting in a set of selected features $G_{i,j}$. We use $G_{i,j}$ to train the subclassifier specialized in discriminating between the two temporal segments in question.

2) Hybrid SVM-HMM: In the second approach, we propose to apply hybrid SVM-HMMs to the problem of AU temporal model detection. Traditionally, HMMs have been used very effectively to model time in classification problems. However, while the sequence of the temporal phases of a facial action over time can be represented very well by HMMs, the HMM suffers from poor discrimination between temporal phases at a single moment in time. The emission probabilities, which are computed for each frame of an input video for the HMM hidden states, are normally modeled by fitting Gaussian mixtures on the features. These Gaussian mixtures are fitted using likelihood maximization, which assumes the correctness of the models (i.e., the feature values should follow a Gaussian distribution) and thus suffers from poor discrimination [76]. Moreover, it results in mixtures trained to model each class and not to discriminate one class from the other.

SVMs, on the other hand, are not suitable for modeling time, but they discriminate extremely well between classes. Using them as emission probabilities might very well result in an improved recognition. We therefore again train one-versus-one SVMs to distinguish the temporal phases neutral, onset, apex, and offset, just as described in Section IV-D1. We then use the output of the component SVMs to compute the emission probabilities. In this way, we arrive at a hybrid SVM-HMM system. This approach has been previously applied with success to speech recognition [77].

HMMs work in a probabilistic framework. On the other hand, the output of an SVM is not a probability measure. The (unsigned) decision function value output $h(\boldsymbol{x})$ of an SVM is a distance measure between a test pattern and the separating hyperplane defined by the support vectors. There is no clear relationship with the posterior class probability $p(y = +1|\boldsymbol{x})$ that the pattern $\boldsymbol{x}$ belongs to the class $y = +1$. However, Platt proposed an estimate for this probability by fitting the SVM output $h(\boldsymbol{x})$ with a sigmoid function [78]

$$p(y = +1|\boldsymbol{x}) = g\left(h(\boldsymbol{x}), A, B\right) \equiv \frac{1}{1 + \exp\left(Ah(\boldsymbol{x}) + B\right)}. \tag{12}$$

The parameters A and B of (12) are found using maximum likelihood estimation of the SVM output on the same data that is used for training each SVM.

As explained in Section IV-D1, we use one-versus-one mc-SVMs to distinguish between temporal phases. This approach is to be preferred over the one-versus-all approach as it aims to learn the solution to a more specific problem, namely, distinguishing between two specific classes. This is in line with our idea of using SVMs for high discriminative power between classes and HMMs to model time.

Our (fully observed) HMM consists of four states, one for each temporal phase. From each SVM, we get, using Platt's

method, pairwise class probabilities $\mu_{ij} \equiv p(c_i|c_i \text{ or } c_j, \boldsymbol{x})$ of the class (HMM state) c_i given the feature vector $\boldsymbol{x}$ and that $\boldsymbol{x}$ belongs to either c_i or c_j. These pairwise probabilities are transformed into posterior probabilities $p(c_i|\boldsymbol{x})$ by

$$p(c_i|\boldsymbol{x}) = 1/\left[\sum_{j=1,j\neq i}^{|C|} \frac{1}{\mu_{ij}} - (|C|-2)\right]. \quad (13)$$

Finally, the posteriors $p(c_i|\boldsymbol{x})$ have to be transformed into *emission probabilities* by applying Bayes' rule

$$p(\boldsymbol{x}|c_i) \propto \frac{p(c_i|\boldsymbol{x})}{p(c_i)} \quad (14)$$

where the *a priori* probability $p(c_i)$ of class c_i is estimated by the relative frequency of the class in the training data.

E. Emotion Detection

To detect the six basic emotions, we use the same set of features, described in Section IV-B. We approach the problem as a dynamic multiclass event detection problem, i.e., for every video, we determine to which class it belongs. To do so, we train an multi-class GentleBoost Support Vector Machines and Hidden Markov Models (MC-GentleSVMs), with a similar structure as the AU temporal segment detector. Again, we train one-versus-one GentleBoost Support Vector Machines (GentleSVMs) to distinguish between pairs of emotions. Because the neutral expression is also present in every video, we also learn classifiers that distinguish between each emotion and the neutral expression. We thus learn 21 binary classifiers. We again use (13) and (14) to determine the emission probabilities used by the SVM. In contrast with the AU temporal segment detector, we do not use the emotions as the state variables, instead we learn the optimal number of states.

V. Utilized Facial Expression Data Sets

In our study, we used four different data sets: the CK-db of volitional facial displays [55], the MMI facial expression database (MMI-db) [79], [80], the DS118 data set of spontaneous facial displays [81], and the triad data set of spontaneous human behavior [82].

The CK-db was developed for research in the recognition of the six basic emotions and their corresponding AUs. The database contains over 2000 near-frontal-view videos of facial displays produced by 210 adults being 18 to 50 years old, 69% female, 81% Caucasian, 13% African, and 6% from other ethnic groups. From this database, 480 gray scale videos have been made publicly available. It is currently the most commonly used database for studies on automatic facial expression analysis. All facial displays were made on command, and the recordings were made under constant lighting conditions. Two certified FACS coders provided AU coding for all videos. Interobserver agreement was expressed in terms of Cohen's kappa coefficient [83], which is the proportion of agreement above what would be expected to occur by chance. The mean kappa for interobserver reliability was 0.82 for AUs at the apex. In the publicly available version of this database, the expressions are shown until the beginning of the apex phase.

The MMI facial expression database has five parts (see [80]). Two FACS experts AU-coded the database. The mean kappa for interobserver reliability was 0.77 for AUs at the apex. The two coders made the final decisions on AU coding by consensus, and this final AU coding was used for the study presented in this paper. The mean kappa for interobserver reliability on Parts I and II of the database was 0.91 for AUs at the apex.

In our study, we use Parts I, II, and IV. Parts I and II contain deliberately displayed facial expressions: 2397 videos depicting facial expressions of single AU activation, multiple AU activations, and six basic emotions. The subjects were 52 adults of 19 to 62 years of age, with 48% being female, 81% Caucasian, 14% Asian, and 5% African. All facial displays were made on command, and the recordings were made under constant lighting conditions from frontal, profile, or dual view orientation. The database contains a large amount of displays of single AU and action descriptor activations. In turn, the MMI data set enables us to learn to recognize every AU independent of other AUs. Part IV of the MMI facial expression database contains currently 65 videos of spontaneous facial displays. Subjects were 18 adults of 21 to 45 years old, with 48% being female, 66% Caucasian, 30% Asian, and 4% African.

To stimulate research into the automatic analysis of AU temporal dynamics, we have made the manual onset–apex–offset coding of Parts I and II publicly available. They can be downloaded from the MMI facial expression database Web site. This will also allow researchers to compare their work against the method proposed here.

The DS118 data set has been collected to study facial expression in patients with heart disease [81]. The subjects were 85 men and women with a history of transient myocardial ischemia who were interviewed on two occasions at a four-month interval. They averaged 59 years of age $(\text{std} = 8.24)$ and were predominantly Caucasian. Spontaneous facial displays were video recorded during a clinical interview that elicited AUs related to disgust, contempt, and other negative emotions as well as smiles. The facial actions displayed in the data are often very subtle. Due to confidentiality issues, this FACS-coded data set is not publicly available. Only the AU coding made by human observers and the tracking data were made available to us.

The triad data set was collected to study the effects of alcohol on the behavior of so-called social drinkers [82]. The subjects were three young Caucasian men, who were recorded simultaneously by three different cameras while drinking and interacting. The recordings are long (over 15 min) and contain displays of diverse facial and bodily gesturing. No AU coding of the data was made publicly available.

VI. Validation Studies

We conducted five sets of experiments to evaluate the performance of different parts of the system: the facial point detector, the facial point tracker, the AU detector, the AU temporal activation model detector, and the six-basic-emotion detector.

TABLE II
AVERAGE CLASSIFICATION RATE OF POINT DETECTION ON MMI FACIAL EXPRESSION DATABASE CK-db

	MMI	CK		MMI	CK
A	0.784	0.920	G	0.982	0.950
A1	0.976	0.960	G1	0.982	0.990
B	0.976	0.960	H	0.976	0.980
B1	0.952	0.990	H1	0.976	0.970
D	0.569	0.960	I	0.904	0.970
D1	0.802	0.950	J	0.928	0.910
E	0.928	0.960	K	0.964	0.930
E1	0.958	0.900	L	0.952	0.800
F	0.982	0.910	M	0.904	0.900
F1	0.982	0.830	N	0.952	0.980
Average for all points:				**0.922**	**0.930**

A. Evaluations of Facial Point Detector

We conducted two experiments to evaluate the performance of our facial point detector: one using the first frames of 300 randomly picked image sequences from the CK-db and the other using the first frames of the 244 image sequences from the MMI-db Part I that will later be used in AU detection. In the experiment with the CK-db images, the proposed facial point detector was evaluated by threefold cross-validation. In the experiment with the MMI-db images, the point detector was trained using all images from the CK-db and tested on the MM-db images. In this way, we were able to test how well the point detector generalizes to entirely different data.

To evaluate the performance of the method, each of the automatically located facial points was compared with the manually annotated point. The error margin was defined in terms of the interocular distance D_I measured in a test image. An automatically detected point displacing e_d pixels from the true facial point is regarded as SUCCESS if $e_d < 0.05D_I$. This means that, e.g., for $D_I = 100$ pixels (a typical value for the CK-db), a bias of up to 5 pixels for an eye corner is regarded as SUCCESS.

Overall, we achieved an average recognition rate of 93% for the samples from the CK-db and 96% for the samples from the MMI-db for 20 facial feature points using the previously described evaluation scheme. The detection rates for each point are given in Table II. The low scores for points D and D1 (the inner eyebrow points) are caused by a slight difference in the definition used during the manual annotation of the two databases: They were labeled slightly beneath the eyebrows for the CK-db and slightly above the eyebrows for the MMI-db.

Facial point detectors developed elsewhere attain 93% to 96% average recognition rate for subsets of the 20 facial points illustrated in Fig. 3 when considering $e_d < 0.3D_I$ as the rule for successful point detection (e.g., [58], [59], and [84]). Hence, the method presented in this work is approximately six times more accurate than the previously reported methods. Typical results of our facial point detector are illustrated in Fig. 4.

B. Evaluations of Facial Point Tracker

We tested the tracking accuracy of the proposed PFFL point tracking algorithm by applying it to several different samples from four different data sets: the CK-db, the MMI Part I and Part II data sets, and the triad data set. We randomly selected 5% of samples from each data set, in such a way that these data are completely independent of the data that we used to model the transition probability models of the tracking algorithm (see Section III). To provide ground truth for our experiments, each frame of each test sequence was labeled by a human observer, provided that all 20 facial points are visible. In case of an occlusion, the location of an occluded point was determined based on its location in the last frame in which the relevant point was visible. The distance metric for a given point $\boldsymbol{p_i}$ is defined per frame as follows:

$$e(\boldsymbol{p_i}, j) = \frac{\|\boldsymbol{p_{i,j}} - \hat{\boldsymbol{p}}_{\boldsymbol{i,j}}\|_2}{D_I(j)} \tag{15}$$

where $D_I(j)$ is the interocular distance, measured at frame j of the test sequence, $\boldsymbol{p_{i,j}}$ is the location of point $\boldsymbol{p_i}$, $i \in [1:20]$ in frame j determined by the tracking algorithm, and $\hat{\boldsymbol{p}}_{i,j}$ is the manually labeled ground truth for the same point at that frame. Fig. 5 shows the mean tracking error for a number of facial points, computed by evaluating the tracking results of 100 videos from the MMI-db. We compute the average error E over all points per frame j as follows:

$$E(j) = \frac{1}{n}\sum_{i=1}^{n} e(\boldsymbol{p_i}, j) \tag{16}$$

where $n = 20$ is the number of points that we track. To determine a classification rate for our tracking result, we use the same measure of success as that which we applied to the point detection results, i.e., a point is tracked successfully in a frame if $E(j) <= 0.05D_I$. Given that the tracking algorithm was trained on samples from the MMI Part I data set (near-frontal views of deliberately displayed facial expressions), it is not surprising that the best results were attained for similar data, i.e., for samples from the CK-db and the MMI Part I data sets, where all points were tracked successfully in 93% and 91% of frames, respectively. On the spontaneous facial data, the tracking algorithm performed less accurately. For the

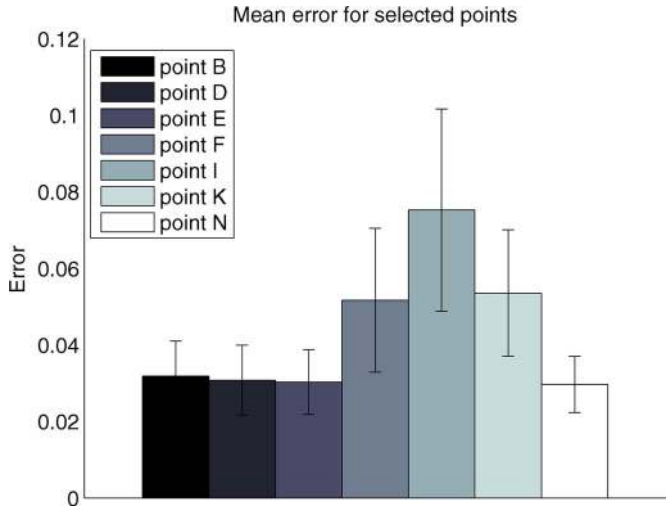

Fig. 5. Mean and standard deviation of the tracking error in units of the interocular distance D_I of selected points. The error is computed over 100 videos taken from the MMI facial expression database.

TABLE III
SUBJECT-INDEPENDENT CROSS-VALIDATION RESULTS FOR AU ACTIVATION DETECTION PER FRAME ON 244 EXAMPLES FROM THE MMI FACIAL EXPRESSION DATABASE

AU	Videos	Frames	Cl. Rate	Recall	Precision	F1
1	22	1006	0.972	0.679	0.728	0.703
2	25	1092	0.961	0.628	0.629	0.628
4	38	1839	0.942	0.582	0.707	0.639
5	19	874	0.949	0.317	0.375	0.344
6	27	1241	0.952	0.695	0.583	0.634
7	15	772	0.963	0.319	0.510	0.392
9	15	636	0.968	0.503	0.477	0.490
10	17	719	0.955	0.266	0.321	0.291
12	17	1004	0.950	0.548	0.482	0.513
13	14	782	0.974	0.668	0.650	0.659
15	15	854	0.944	0.412	0.344	0.375
16	18	717	0.947	0.230	0.229	0.229
18	16	568	0.974	0.593	0.523	0.556
20	15	871	0.964	0.696	0.554	0.617
22	15	696	0.964	0.536	0.467	0.499
24	15	536	0.955	0.497	0.503	0.500
25	105	5401	0.909	0.810	0.831	0.821
26	32	1597	0.875	0.198	0.179	0.188
27	15	800	0.983	0.720	0.819	0.766
30	15	736	0.972	0.438	0.588	0.502
43	15	750	0.973	0.520	0.657	0.580
45	107	1243	0.956	0.668	0.625	0.645
46	6	130	0.913	0.723	0.667	0.694
Avg:			**0.953**	**0.532**	**0.541**	**0.533**

TABLE IV
SUBJECT-INDEPENDENT CROSS-VALIDATION RESULTS FOR AU ACTIVATION DETECTION PER FRAME ON 153 EXAMPLES FROM THE CK-db

AU	Videos	Frames	Cl. Rate	Recall	Precision	F1
1	68	883	0.918	0.808	0.844	0.826
2	50	657	0.939	0.791	0.879	0.833
4	54	857	0.870	0.604	0.658	0.630
5	37	421	0.904	0.566	0.629	0.596
6	39	535	0.930	0.789	0.811	0.800
7	31	415	0.870	0.268	0.315	0.290
9	30	357	0.928	0.676	0.497	0.573
10	26	302	0.914	0.403	0.401	0.402
12	42	727	0.930	0.827	0.844	0.836
15	19	264	0.969	0.500	0.283	0.361
20	34	381	0.908	0.466	0.582	0.517
24	17	297	0.935	0.395	0.497	0.440
25	19	1572	0.851	0.717	0.782	0.748
26	27	344	0.902	0.336	0.380	0.357
27	30	800	0.964	0.836	0.873	0.854
45	23	1243	0.943	0.584	0.408	0.480
Avg:			**0.917**	**0.598**	**0.605**	**0.596**

MMI Part II data set and the triad data set, the tracking of all points was successful in 77% and 52% of frames, respectively. Note, however, that samples from both of these data sets of spontaneous facial data contain instances of occluded facial points, which had a large influence on the average distance metric $E(j)$.

C. Frame-Based AU Detection Evaluation

We tested our AU detector system on both the MMI-db and the CK-db, measuring for each frame of a video whether it was correctly classified as containing an active AU or not (regardless of the temporal phase). On the MMI-db, we tested it for all 22 AUs that can be detected using a geometric-feature-based approach. The set was created so that it includes for every AU at least 15 examples. For AU13 (a smile with the mouth corners sharply pulled upward), we could find only 14 examples, and for AU46 (wink), we could find only 6. Some AUs always occur in combination with others. For instance, AU22, which puffs the lips as in pronouncing the word "flirt," will always cause the lips to part and thus to display AU25. Thus, for some AUs, we have more occurrences than for others. In the CK-db, not all 22 target AUs are present in sufficient numbers. Hence, we have tested our AU detector on the CK-db only for those AUs that were present with at least 15 examples.

All studies were performed by leave-one-subject-out cross-validation, which ensures that we train and evaluate a subject-independent system. Results for the MMI-db are shown in Table III, and those for the CK-db are shown in Table IV. The number of videos in which each AU occurs is listed in the second column of the tables, and the total number of frames in which an AU is active is given in the third column. For comparison with older works, we show the classification rate in the fourth column. Because of the highly unbalanced nature of our data, this performance measure is overly optimistic. More detailed frame-based AU detection performance results are provided in terms of ROC curves in Fig. 6.

Although precision and recall are better measures of performance when dealing with unbalanced data sets, it is difficult to compare performances using two measures. Therefore, we have also included the F1 measure, which favors precision (p) and recall (r) equally. The F1 measure is defined as $2pr/(p+r)$. The results show that the AUs 1, 2, 4, 6, 12, 13, 18, 20, 25, 27, 30, 43, 45, and 46 are detected well. AU5 and AU7 both involve only the movements of the upper and the lower eyelid. The eyelids move up or down only very little when these AUs are activated, and we believe that our tracker is not sensitive enough to attain highly accurate results for these AUs. AU26 (jaw dropped) is very similar to AU27 (mouth stretched open). In fact, in an activation of AU27, the facial points around the mouth will go through all the positions that they would go through in case of AU26 activation. Therefore, the two AUs are hard to separate. Similarly, AU10 and AU16 are characterized by point displacements that are very similar to point displacements caused by other AUs that also raise the upper lip (AU10) or lower the lower lip (AU16).

D. AU Temporal Model Detection Evaluation

We evaluated the performance of our temporal model detector on examples from the MMI-db only. This is because the CK-db videos were cut after the expressions reached the apex phase. Therefore, they do not display the full temporal model of facial expressions. Fig. 7 compares the F1 measures attained by the two tested approaches (see Section IV-D): multi-class GentleBoost Support Vector Machines only and the hybrid GentleBoost Support Vector Machines approach. The accuracy was measured per frame (i.e., for each frame, we checked whether it was assigned the correct phase label).

We see that, compared with the multiclass GentleSVM method, the detection of the apex phase has benefited most from introducing the HMM. The apex phase had an increase in F1

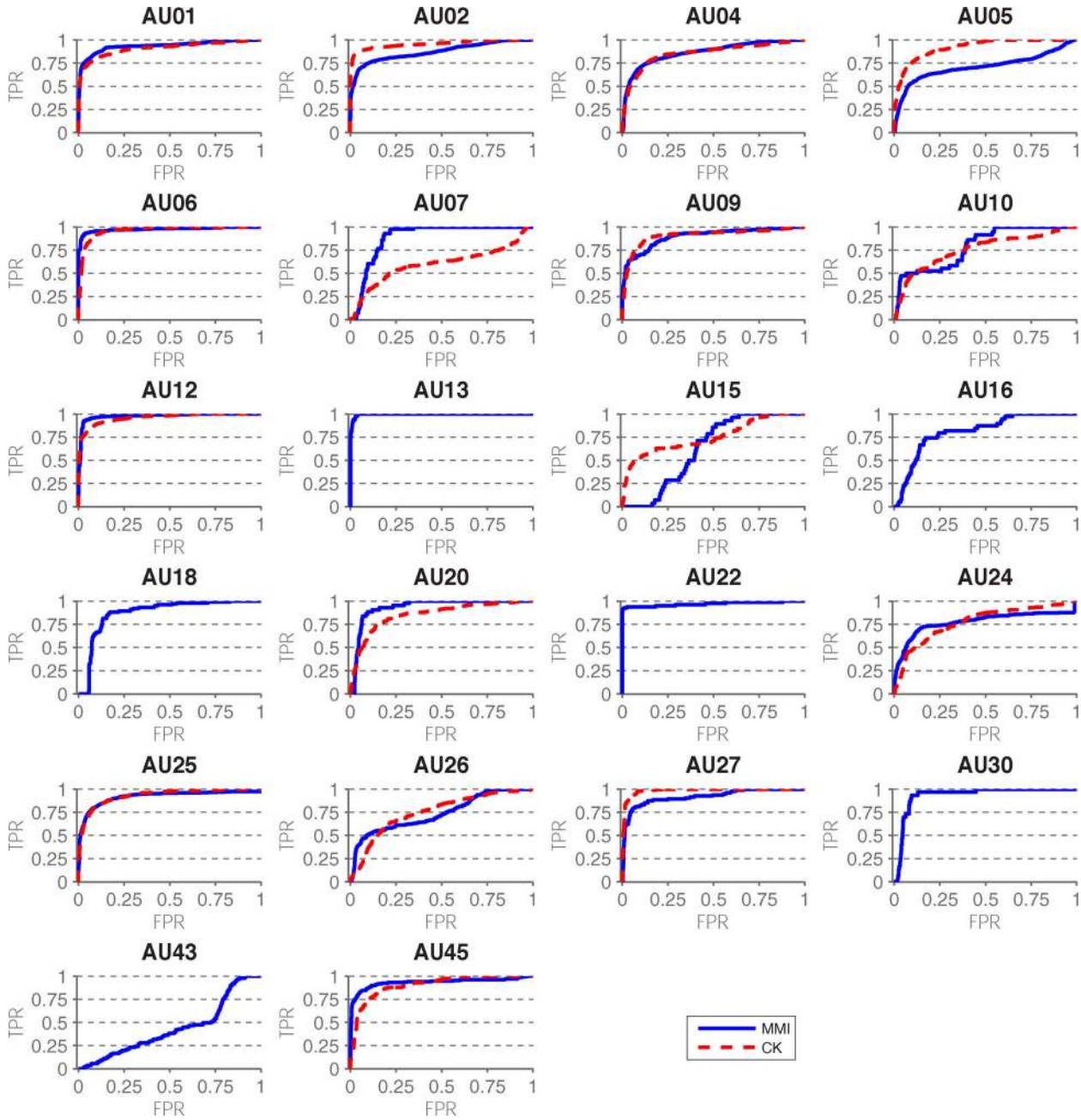

Fig. 6. ROC curves of AU activation detection per frame on the MMI and the CK data sets. For AU13, AU16, AU18, AU22, AU30, and AU43, the CK data set did not contain enough examples to perform AU detection.

of 8%, the offset 6.8%, the onset phase 3.6%, and the neutral phase 3.4% (relative to mc-SVM). The fact that the neutral phase benefits least from the addition of the HMM is expected because this is not a dynamic part of the facial action. The effect of applying the grammatical rules is less successful. While it attains good results for the offset phase and, in a limited way, for the neutral phase, it actually decreases the accuracy of the onset and apex phase recognition.

Detailed results per AU are shown for the SVM-HMM approach in Table V. Fig. 8 shows one example of the recognition of the temporal phases of a video containing an AU 25 activation. The figure shows that the prediction (red dotted line) is one frame late at predicting the first and second apex phases. It also predicts the last offset phase to stop six frames too early. The SVM-HMM system did recognize correctly that there are two apex phases.

We also looked into the durations of the facial actions, both the total duration of an AU (i.e., the number of consecutive frames that were predicted to be nonneutral) and the durations of the temporal phases separately. Fig. 9 shows the statistics for this analysis. The duration error is measured in frames. The figure shows the average number of frames that a temporal phase duration or the entire AU activation duration is off, averaged per AU. We can see that, for most AUs, the average error per temporal phase is less than four frames. The apex temporal phase has the largest error. We can also see from Fig. 9 that the error of the total AU activation duration is far less than the sum of the temporal phase duration errors. This is because,

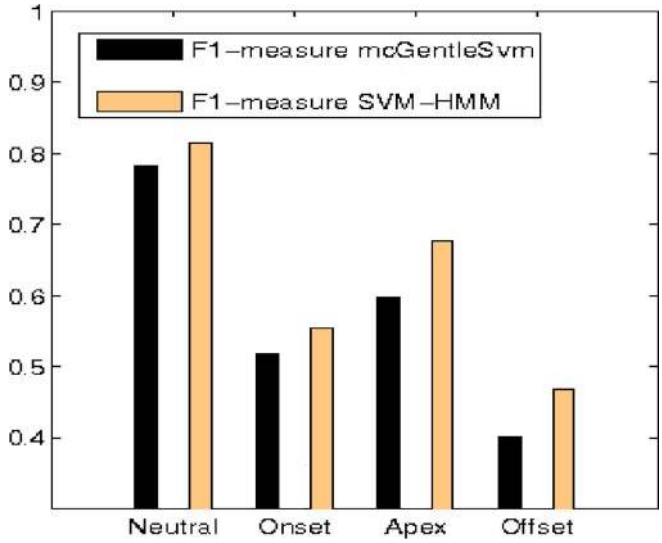

Fig. 7. Comparison of the F1 measures attained by the two temporal model detector approaches, measured per frame.

TABLE V
F1-MEASURE CLASSIFICATION ACCURACY OF HYBRID SVM-HMM FOR DISTINGUISHING THE FOUR TEMPORAL PHASES

AU	Neutral	Onset	Apex	Offset
1	0.790	0.669	0.585	0.536
2	0.848	0.544	0.730	0.642
4	0.690	0.521	0.615	0.334
5	0.610	0.352	0.561	0.292
6	0.807	0.469	0.693	0.374
7	0.784	0.100	0.390	0.108
9	0.895	0.756	0.887	0.462
10	0.855	0.587	0.790	0.323
12	0.931	0.693	0.773	0.679
13	0.926	0.847	0.750	0.642
15	0.791	0.339	0.742	0.357
16	0.815	0.481	0.600	0.384
18	0.914	0.569	0.740	0.592
20	0.883	0.734	0.860	0.583
22	0.864	0.701	0.469	0.373
24	0.507	0.257	0.547	0.037
25	0.865	0.634	0.776	0.631
26	0.751	0.490	0.583	0.417
27	0.720	0.747	0.858	0.708
30	0.787	0.461	0.541	0.415
43	0.937	0.476	0.758	0.728
45	0.971	0.780	0.653	0.710
46	0.618	0.146	0.182	0.239
Avg:	**0.807**	**0.537**	**0.656**	**0.459**

usually, if the apex phase has been predicted to last too long, consequently, the offset phase will start late and results in an error in the offset phase duration; thus, the error is effectively double counted.

E. Event-Based AU Detector Evaluation

Aside from AU detection per frame, we also want to be able to perform the so-called event coding, i.e., we want to determine which AUs were active in an entire image sequence.

1) Within-Database Evaluation: The simplest way to perform event detection is to use a threshold on the number of frames predicted active by the frame-based AU detector. As the SVM classifier adds an *a priori* unknown amount of noise to its output in the form of false positives and false negatives, fixing a threshold based on, for example, the minimal duration of an AU as observed by psychologists will not necessarily achieve optimal results. To overcome this problem, we add a decision layer that will empirically learn a threshold θ based on the AUs automatically detected per frame.

Another way to determine whether an AU was present in a video is to analyze the output of the AU temporal model detector. When doing this, we regard an AU to be present if the temporal model detector predicted a correct sequence of phases (e.g., neutral $\rightarrow$ onset $\rightarrow$ apex $\rightarrow$ offset). Table VI compares the AU event detection results of the simple threshold-based method with that of the temporal model detector method, where we have used the SVM-HMM approach to detect the temporal phases of each AU. As the table shows, using the hybrid SVM-HMM method for AU event detection results in a 17.1% improvement in F1 measure, clearly showing the benefit of this approach.

2) Cross-Database Evaluation: A cross-validation study on data from a single database might attain very good results, but it does not guarantee that the evaluated system performs well on novel data. To test the generalizability of the results, we train the system on data from one database and test it on data from a second database. Both databases must be recorded completely independently of each other. That exactly is the case for the MMI-db and the CK-db.

We performed two tests. In the first experiment, we train the AU detector on all data from the MMI-db and test it on data from the CK-db. Vice versa, in the second experiment, we train on the CK-db and test on the MMI-db. The results, measured per image sequence (event detection) in terms of the F1 measure, are shown in Table VII. The performance of the MMI-trained system is almost 10% higher than that of the CK-trained system. We believe that this is due to the low variability of facial expressions in the latter database. As AUs in the CK-db occur frequently in very similar configurations (e.g., AU1 + AU2 + AU5 + AU25 + AU27 for the expression of surprise), an AU detection system trained on this data will expect AUs to be produced in a similar fashion in the test examples. However, this is not the case for the MMI-db data, where individual AU activations often occur. On the other hand, we see that the MMI-trained system generalizes reasonably well on data from a completely different database, although the F1 measure is still a good 22% lower than that attained when performing event detection within the MMI-db (see Table VI), and thus, high generalization has not yet been achieved.

3) Spontaneous Data Evaluation: The AU detection evaluations presented so far were performed on acted data. That is, the expressions shown in the data were produced on command. Spontaneous expressions, however, are different both in their composition of AUs as well as in their temporal dynamics [85]. Ultimately, we would like to deploy our facial expression analysis system in such real-world situations. As mentioned in the introduction, very few works have focused on the problem of spontaneous facial expression recognition so far, and results have been quite limited (see [8] for an overview).

We tested our system on the DS118, Triad, and MMI-db part II databases (see Section V). We trained the system on all

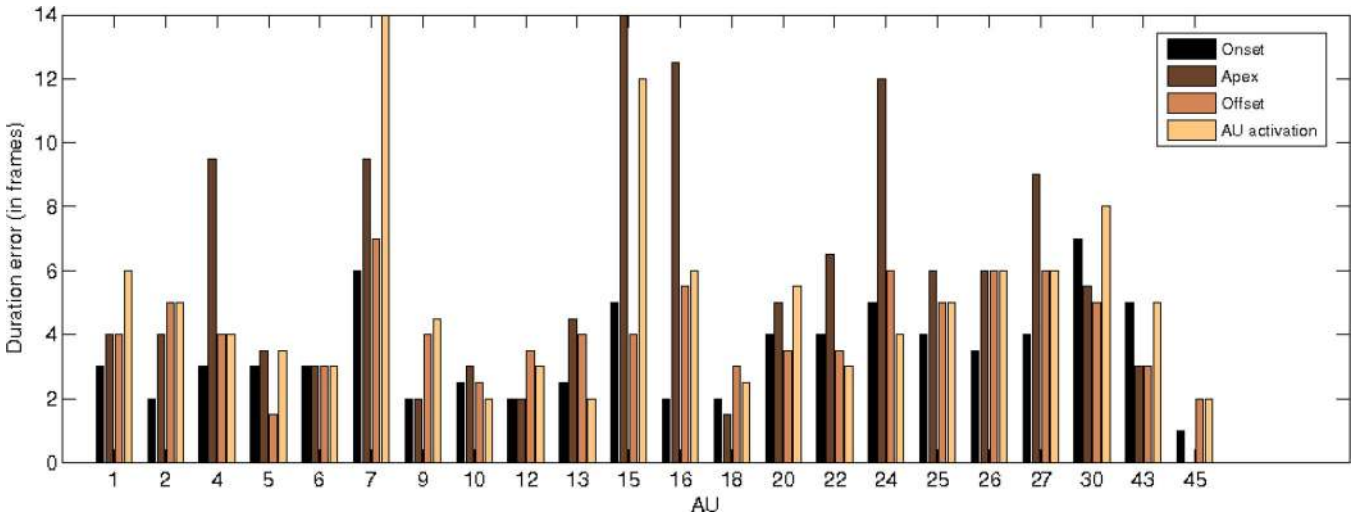

Fig. 8. Example of temporal phase recognition for AU25. The solid line shows the ground truth labeling per frame, and the dotted line shows the prediction by the SVM-HMM. Horizontal lines depict either a neutral or an apex phase, upward slopes an onset phase, and downward slopes an offset phase.

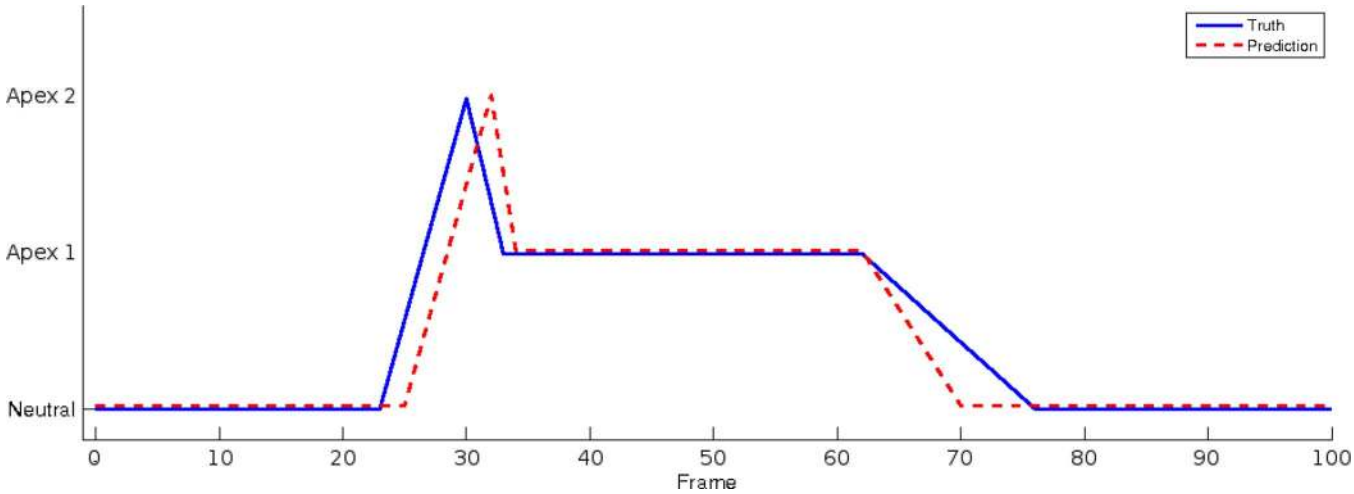

Fig. 9. Temporal segment (onset, apex, and offset) duration error and the entire facial action duration error. Results are averaged per AU and measured in frames.

TABLE VI
COMPARISON OF AU EVENT DETECTION METHODS ON THE MMI FACIAL EXPRESSION DATABASE

System	Cl. Rate	Recall	Precision	F1
Threshold approach	0.876	0.772	0.402	0.521
Hybrid SVM-HMM	0.943	0.756	0.653	0.692

TABLE VII
F1 MEASURE FOR CROSS-DATABASE AU DETECTION PER VIDEO. SYSTEM WAS EITHER TRAINED ON 244 EXAMPLES FROM THE MMI FACIAL EXPRESSION DATABASE AND TESTED ON 153 EXAMPLES FROM THE CK-db OR VICE VERSA

AU	Train MMI	Train CK	AU	Train MMI	Train CK
1	0.661	0.255	12	0.635	0.400
2	0.762	0.467	15	0.372	0.229
4	0.541	0.414	20	0.277	0.341
5	0.447	0.149	24	0.333	0.292
6	0.429	0.571	25	0.799	0.746
7	0.129	0.211	26	0.293	0.203
9	0.495	0.286	27	0.589	0.591
10	0.232	0.109	45	0.442	0.622
Average results:				**0.465**	**0.368**

available posed data from the MMI-db part I. On spontaneous data of smiles, taken from the triad database and the MMI-db part II, AU6 was recognized correctly 77% of the times, AU12 in 54%, and AU13 in 85% of the videos. The reason why AU12 has a rather low classification rate is that AU12 and AU13 are very similar. Both involve movement of the mouth corners. The difference lies in the horizontal movement: With AU12, the mouth corners move further out while, with AU13, the mouth corners are pulled up sharply.

TABLE VIII
CONFUSION MATRIX OF EMOTION DETECTION ON THE CK-db. ROWS INDICATE GROUND TRUTH, AND COLUMNS INDICATE DETECTED EMOTIONS

	ANGR	DISG	FEAR	HAPP	SADN	SURP
ANGR	2	3	2	0	9	1
DISG	1	19	1	1	4	1
FEAR	1	4	15	5	2	1
HAPP	1	0	3	33	0	1
SADN	4	2	1	0	16	1
SURP	0	1	1	1	0	34
Cl. rate	0.118	0.704	0.536	0.868	0.667	0.919

On the DS118 database, we tested for brow-related AUs only (i.e., AU1, AU2, and AU4). We achieved a 50.4% classification rate for AU event detection (i.e., detecting the presence of an AU within a video). Although this is not a very high result, it is promising considering that we were not able to use any

TABLE IX
COMPARISON OF CLASSIFICATION RATE OF EXISTING WORKS THAT REPORT AU DETECTION ON EITHER THE MMI FACIAL EXPRESSION DATABASE OR CK-db. FOURTH COLUMN INDICATES IF THE SYSTEM IS CAPABLE OF DETECTING THE TEMPORAL PHASES OF AN AU

System	feature type	Classification method	temporal	AUs	videos MMI	Cl. Rate MMI	videos CK	Cl. Rate CK
This work	Geometric	GentleSvm & HMM	1	22	244	**0.953**	153	0.917
Bartlett et al. 2006 [8]	Appearance	GentleSvm	0	19	–	–	Unknown	0.909
Koelstra and Pantic 2008 [43]	Appearance	GentleBoost HMM	1	27	264	0.943	143	0.891
Littlewort et al. 2006 [48]	Appearance	GentleSvm	0	7	Unknown	0.927	–	–
Pantic and Patras 2005 [56]	Geometric	Rule-based	1	27	299	0.936	–	–
Tian et al. 2001 [69]	Hybrid	ANN	0	18	–	–	465	**0.950**
Tong et al. 2007 [71]	Appearance	AdaBoost & DBN	0	14	–	–	Unknown	0.933
Whitehill and Omlin [81]	Appearance	AdaBoost	0	11	–	–	Unknown	0.925
Yang et al. 2009 [84]	Appearance	AdaBoost	0	8	–	–	Unknown	Unknown

spontaneous training data. Other researchers reported between 26% [35] and 76% [34] classification rate for brow actions in widely varying data sets.

F. Emotion Detection Evaluation

The detection of six basic emotions in posed facial expression databases is considered to be largely solved, particularly when the subject being tested is known and was part of the training data. However, for optimal comparability with existing automatic facial expression recognition works, we evaluate our six-basic-emotion detection system on 171 videos taken from the CK-db. The videos were selected with the criterion that two coders were able to attain a consensus on what emotion was shown in that video. This is a stricter ground truth criterion than using the ground truth provided with the CK-db. This strategy was used to reduce the label error in the data set.

Table VIII shows the confusion matrix and classification rates of all emotions. Emotions are detected per video, i.e., the table shows event detection results. From the results, we have to conclude that it is very hard to distinguish between the emotions angry and sadness. The reason for this is that both expressions often incur similar brow movements. From a geometric point point of view, the difference is in the downward motion of the lip corners, and unfortunately, that motion can be very subtle. It also shows that fear is often confused with either disgust or happiness. While the confusion with fear is common, the confusion with happiness is somewhat surprising. Again, the explanation lies in the displacement of the lip corners. The motion of the lip corners caused by AU20 and AU12 can be quite similar, particularly if the tracking is slightly off. We believe that this is an indication that four points is insufficient to capture the different shapes of the mouth. Moving toward eight or more points would allow a geometric-based approach to better distinguish between AU12, AU20, and AU15, which we believe are the main culprits in the confusions made by our emotion detection system.

G. Performance Comparison With Previous Works

Although there is still no standardized method for the evaluation of automatic facial expression recognition systems, many works have reported the performance of their system on one or more publicly available databases. More specifically, many works have used the CK-db [55], the MMI facial expression database [79], or both. Therefore, a comparison is possible to a certain extent, although using the same database does not guarantee that the systems were trained and tested with the same number of videos from each database nor does it guarantee that the same rules, e.g., for the optimization of parameters, were adhered to.

Table IX gives an overview of the existing systems that report their performance in terms of AU event detection on either the CK-db, the MMI facial expression database, or both. For [86], we are unable to report a classification rate on either databases, as the authors only mention the achieved area under the ROC curve in their paper. As we can see, our proposed approach outperforms all other methods on the MMI facial expression database, and of the methods capable of detecting temporal segments, it also scores the highest on the CK-db. Although this is not a comparison in a controlled experiment, it still shows that the proposed system performs well compared to existing approaches. It also shows that appearance-based approaches do not necessarily outperform geometric-feature-based approaches.

VII. CONCLUSION

Accurate fully automatic facial expression analysis would have many real-world applications. In this paper, we have shown that not only fully automatic highly accurate AU activation detection based on geometric features is possible but also that it is possible to detect the four temporal phases of an AU with high accuracy and that geometric features are very well suited for this task. The proposed system was tested extensively on multiple databases and was shown to generalize well when trained on data from one database and tested on data from another. This being said, generalization to completely novel data is not possible yet without some loss of accuracy. At this point, a major limitation of the system is that it can only recognize facial expressions as long as the face is viewed from a pseudofrontal view. If the head has an out-of-plane rotation greater than 20°, the system will fail. This is something that we wish to address in our future research.

ACKNOWLEDGMENT

The authors would like to thank J. Cohn of the University of Pittsburgh for providing the Cohn–Kanade database, as well as the ground truth and the tracking data for the part of the DS118 data set used in this study. The authors would also like to thank M. Sayette of the University of Pittsburgh for providing the triad data set and I. Patras of Queen Mary University London for his valuable comments.

REFERENCES

[1] P. Ekman and E. L. Rosenberg, *What the Face Reveals: Basic and Applied Studies of Spontaneous Expression Using the Facial Action Coding System*. Oxford, U.K.: Oxford Univ. Press, 2005.

[2] J. Russell and J. Fernandez-Dols, *The Psychology of Facial Expression*. New York: Cambridge Univ. Press, 1997.

[3] B. Golomb and T. Sejnowski, "Benefits of machine understanding of facial expressions," in *NSF Report—Facial Expression Understanding*, P. Ekman, T. Huang, T. Sejnowski, and J. Hager, Eds. Salt Lake City, UT, 1997, pp. 55–71.

[4] M. Pantic, "Face for ambient interface," in *Ambient Intelligence in Everyday Life*, vol. 3864, *Lecture Notes on Artificial Intelligence*. Berlin, Germany: Springer-Verlag, 2006, pp. 32–66.

[5] M. Cohen, *Perspectives on the Face*. Oxford, U.K.: Oxford Univ. Press, 2006.

[6] A. Young, *Face and Mind*. Oxford, U.K.: Oxford Univ. Press, 1998.

[7] M. Pantic and L. J. M. Rothkrantz, "Toward an affect-sensitive multimodal human–computer interaction," *Proc. IEEE*, vol. 91, no. 9, pp. 1370–1390, Sep. 2003.

[8] Z. Zeng, M. Pantic, G. Roisman, and T. Huang, "A survey of affect recognition methods: Audio, visual and spontaneous expressions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 1, pp. 39–58, Jan. 2009.

[9] Y. L. Tian, T. Kanade, and J. F. Cohn, *Handbook of Face Recognition*. New York: Springer-Verlag, 2005.

[10] M. Pantic and M. Bartlett, "Machine analysis of facial expressions," in *Face Recognition*, K. Delac and M. Grgic, Eds. Vienna, Austria: I-Tech Educ. Publishing, 2007, pp. 377–416.

[11] J. F. Cohn, "Foundations of human computing: Facial expression and emotion," in *Proc. ACM Int. Conf. Multimodal Interfaces*, 2006, vol. 1, pp. 610–616.

[12] P. Ekman, *Face of Man: Universal Expression in a New Guinea Village*. New York: Garland, 1982.

[13] W.-F. Liu, J.-L. Lu, Z.-F. Wang, and H.-J. Song, "An expression space model for facial expression analysis," in *Proc. CISP*, May 2008, vol. 2, pp. 680–684.

[14] K. Anderson and P. McOwan, "A real-time automated system for recognition of human facial expressions," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 36, no. 1, pp. 96–105, Feb. 2006.

[15] S. Park, J. Shin, and D. Kim, "Facial expression analysis with facial expression deformation," in *Proc. 19th ICPR*, Dec. 2008, pp. 1–4.

[16] I. Kotsia and I. Pitas, "Facial expression recognition in image sequences using geometric deformation features and support vector machines," *IEEE Trans. Image Process.*, vol. 16, no. 1, pp. 172–187, Jan. 2007.

[17] R. E. Kaliouby and P. Robinson, "Real-time inference of complex mental states from facial expressions and head gestures," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2004, vol. 3, p. 154.

[18] A. Ashraf, S. Lucey, T. Chen, J. Cohn, and K. Prkachin, "The painful face—Pain expression recognition using active appearance models," in *Proc. Int. Conf. Multimedia Interfaces*, 2007, pp. 9–14.

[19] G. Littlewort, M. Bartlett, and K. Lee, "Faces of pain: Automated measurement of spontaneous facial expressions of genuine and posed pain," in *Proc. Int. Conf. Multimodal Interfaces*, 2007, pp. 15–21.

[20] H. Gu and Q. Ji, "Information extraction from image sequences of real-world facial expressions," *Mach. Vis. Appl.*, vol. 16, no. 2, pp. 105–115, Feb. 2005.

[21] J. Cohn and P. Ekman, "Measuring facial action by manual coding, facial EMG, and automatic facial image analysis," in *Handbook of Nonverbal Behavior Research Methods in the Affective Sciences*, J. A. Harrigan, R. Rosenthal, and K. Scherer, Eds. New York: Oxford Univ. Press, 2005, pp. 9–64.

[22] P. Ekman, W. V. Friesen, and J. C. Hager, *Facial Action Coding System*. Salt Lake City, UT: A Human Face, 2002.

[23] D. Cunningham, M. Kleiner, C. Wallraven, and H. Bülthoff, "The components of conversational facial expressions," in *Proc. ACM Int. Symp. Appl. Perception Graph. Vis.*, 2004, pp. 143–149.

[24] A. C. de C. Williams, "Facial expression of pain: An evolutionary account," *Behav. Brain Sci.*, vol. 25, no. 4, pp. 439–488, Aug. 2002.

[25] P. Ekman and W. Friesen, "The repertoire of nonverbal behavioral categories—Origins, usage and coding," *Semiotica*, vol. 1, pp. 49–98, 1969.

[26] M. F. Valstar and M. Pantic, "Biologically vs. logic inspired encoding of facial actions and emotions in video," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2006, pp. 325–328.

[27] M. S. Bartlett, P. Viola, T. Sejnowski, B. Golomb, J. Larsen, J. Hager, and P. Ekman, "Classifying facial actions," in *Proc. Adv. Neural Inf. Process. Syst.*, 1996, vol. 8, pp. 823–829.

[28] J. Lien, T. Kanade, J. Cohn, and C. Li, "Subtly different facial expression recognition and expression intensity estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 1998, pp. 853–859.

[29] M. Pantic, L. J. M. Rothkrantz, and H. Koppelaar, "Automation of nonverbal communication of facial expressions," in *Proc. Conf. Euromedia*, 1998, pp. 86–93.

[30] M. S. Bartlett, J. C. Hager, P. Ekman, and T. J. Sejnowski, "Measuring facial expressions by computer image analysis," *Psychophysiology*, vol. 36, no. 2, pp. 253–263, Mar. 1999.

[31] Y. Tian, T. Kanade, and J. Cohn, "Recognizing action units for facial expression analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 2, pp. 97–115, Feb. 2001.

[32] M. Pantic and L. Rothkrantz, "Facial action recognition for facial expression analysis from static face images," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 34, no. 3, pp. 1449–1461, Jun. 2004.

[33] M. Pantic and I. Patras, "Dynamics of facial expressions—Recognition of facial actions and their temporal segments from face profile image sequences," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 36, no. 2, pp. 433–449, Apr. 2006.

[34] J. F. Cohn, J. F. Reed, Z. Ambadar, J. Xiao, and T. Moriyama, "Automatic analysis and recognition of brow actions in spontaneous facial behavior," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, 2004, vol. 1, pp. 610–616.

[35] M. S. Bartlett, G. Littlewort, M. G. Frank, C. Lainscsek, I. Fasel, and J. Movellan, "Fully automatic facial action recognition in spontaneous behavior," in *Proc. IEEE Int. Conf. Autom. Face Gesture Recog.*, 2006, pp. 223–230.

[36] M. F. Valstar, M. Pantic, Z. Ambadar, and J. F. Cohn, "Spontaneous vs. posed facial behavior: Automatic analysis of brow actions," in *Proc. ACM Int. Conf. Multimodal Interfaces*, 2006, pp. 162–170.

[37] M. F. Valstar, M. Pantic, and H. Gunes, "A multimodal approach to automatic recognition of posed vs. spontaneous smiles," in *Proc. ACM Int. Conf. Multimodal Interfaces*, 2007, pp. 38–45.

[38] Z. Ambadar, J. Schooler, and J. F. Cohn, "Deciphering the enigmatic face: The importance of facial dynamics in interpreting subtle facial expressions," *Psychol. Sci.*, vol. 16, no. 5, pp. 403–410, May 2005.

[39] L. Gralewski, N. Campbell, and I. Voak, "Using a tensor framework for the analyisis of facial dynamics," in *Proc. IEEE Int. Conf. Autom. Face Gesture Recog.*, 2006, pp. 217–222.

[40] Y. Tong, W. Liao, and Q. Ji, "Facial action unit recognition by exploiting their dynamic and semantic relationships," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 10, pp. 1683–1699, Oct. 2007.

[41] Y. Zhang and Q. Ji, "Active and dynamic information fusion for facial expression understanding from image sequence," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 5, pp. 699–714, May 2005.

[42] G. Littlewort, M. Bartlett, I. Fasel, J. Susskind, and J. Movellan, "Dynamics of facial expression extracted automatically from video," *Image Vis. Comput.*, vol. 24, no. 6, pp. 615–625, Jun. 2006.

[43] S. Koelstra and M. Pantic, "Non-rigid registration using free-form deformations for recognition of facial actions and their temporal dynamics," in *Proc. IEEE Int. Conf. Autom. Face Gesture Recog.*, 2008, pp. 1–8.

[44] M. Pantic and I. Patras, "Detecting facial actions and their temporal segments in nearly frontal-view face image sequences," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, 2005, pp. 3358–3363.

[45] M. F. Valstar, I. Patras, and M. Pantic, "Facial action unit detection using probabilistic actively learned support vector machines on tracked facial point data," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2005, vol. 3, pp. 76–84.

[46] M. F. Valstar and M. Pantic, "Fully automatic facial action unit detection and temporal analysis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2006, p. 149.

[47] M. F. Valstar and M. Pantic, "Combined support vector machines and hidden Markov models for modeling facial action temporal dynamics," in *Proc. IEEE Workshop Human Comput. Interaction*, vol. 4796, *Lecture Notes on Computer Science*, 2007, pp. 118–127.

[48] G. Donato, M. Bartlett, J. Hager, P. Ekman, and T. Sejnowski, "Classifying facial actions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 10, pp. 974–989, Oct. 1999.

[49] B. Jiang, M. Valstar, and M. Pantic, "Action unit detection using sparse appearance descriptors in space-time video volumes," in *Proc. IEEE Int. Conf. Autom. Face Gesture Recog.*, 2011, pp. 314–321.

[50] E. Hjelmås and B. Low, "Face detection: A survey," *Comput. Vis. Image Understand.*, vol. 83, no. 3, pp. 236–274, 2001.

[51] M. Yang, D. Kriegman, and N. Ahuja, "Detecting faces in images: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 1, pp. 34–58, Jan. 2002.

[52] S. Li and A. Jain, *Handbook of Face Recognition*. New York: Springer-Verlag, 2005.

[53] P. Viola and M. Jones, "Robust real-time object detection," *Int. J. Comput. Vis.*., vol. 57, no. 2, pp. 137–154, May 2004.
[54] I. Fasel, B. Fortenberry, and J. Movellan, "A generative framework for real time object detection and classification," *Comput. Vis. Image Understand.*, vol. 98, no. 1, pp. 181–210, Apr. 2005.
[55] T. Kanade, J. Cohn, and Y. Tian, "Comprehensive database for facial expression analysis," in *Proc. IEEE Int. Conf. Autom. Face Gesture Recog.*, 2000, pp. 46–53.
[56] D. Vukadinovic and M. Pantic, "Fully automatic facial feature point detection using Gabor feature based boosted features," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, 2005, pp. 1692–1698.
[57] E. Holden and R. Owens, "Automatic facial point detection," in *Proc. Asian Conf. Comput. Vis.*, 2002, pp. 731–736.
[58] L. Chen, L. Zhang, H. Zhang, and M. Abdel-Mottaleb, "3D shape constraint for facial feature localization using probabilistic-like output," in *Proc. IEEE Int. Workshop Anal. Model. Faces Gestures*, 2004, pp. 302–307.
[59] D. Cristinacce and T. Cootes, "A comparison of shape constrained facial feature detectors," in *Proc. IEEE Int. Conf. Autom. Face Gesture Recog.*, 2004, pp. 375–380.
[60] D. Cristinacce and T. Cootes, "Facial feature detection and tracking with automatic template selection," in *Proc. IEEE Int. Conf. Autom. Face Gesture Recog.*, 2006, pp. 429–434.
[61] S. Baker, D. Scharstein, J. P. Lewis, S. Roth, M. Black, and R. Szeliski, "A database and evaluation methodology for optical flow," *Int. J. Comput. Vis.*, vol. 92, no. 1, pp. 1–31, Mar. 2011.
[62] J. Barron, D. Fleet, and S. Beauchemin, "Performance of optical flow techniques," *Int. J. Comput. Vis.*, vol. 12, no. 1, pp. 43–78, Feb. 1994.
[63] B. McCane, K. Novins, D. Crannitch, and B. Galvin, "On benchmarking optical flow," *Comput. Vis. Image Understand.*, vol. 84, no. 1, pp. 126–143, Oct. 2001.
[64] B. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proc. Int. Joint Conf. Artif. Intell.*, 1981, pp. 674–679.
[65] J. Xiao, S. Baker, I. Matthews, and T. Kanade, "Real-time combined 2D+3D active appearance models," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, 2004, vol. 2, pp. 535–542.
[66] R. Kalman, "A new approach to linear filtering and prediction problems," *Trans. ASME, J. Basic Eng.*, vol. 82, pp. 35–45, 1960.
[67] C. Andrieu, A. Doucet, S. Singh, and V. Tadic, "Particle methods for change detection, system identification, and control," *Proc. IEEE*, vol. 92, no. 3, pp. 423–438, Mar. 2004.
[68] S. Hamlaoui and F. Davoine, "Facial action tracking using particle filters and active appearance models," in *Proc. IEEE Int. Conf. Face Gesture Recog.*, 2005, pp. 165–169.
[69] J. McCall and M. Trivedi, "Facial action coding using multiple visual cues and a hierarchy of particle filters," in *Proc. Int. Conf. Comput. Vis. Pattern Recog.*, 2006, vol. 3, p. 150.
[70] I. Patras and M. Pantic, "Particle filtering with factorized likelihoods for tracking facial features," in *Proc. Int. Conf. Autom. Face Gesture Recog.*, 2004, pp. 97–102.
[71] M. K. Pitt and N. Shephard, "Filtering via simulation: Auxiliary particle filters," *J. Amer. Stat. Assoc.*, vol. 94, no. 446, pp. 590–599, Jun. 1999.
[72] I. Patras and M. Pantic, "Tracking deformable motion," in *Proc. Int. Conf. Syst., Man, Cybern.*, 2005, pp. 1066–1071.
[73] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernel-Based Learning*. New York: Cambridge Univ. Press, 2000.
[74] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: A statistical view of boosting," *Ann. Statist.*, vol. 28, no. 2, pp. 337–374, 2000.
[75] J. F. Cohn and K. L. Schmidt, "The timing of facial motion in posed and spontaneous smiles," *J. Wavelets, Multi-Resolution Inf. Process.*, vol. 2, no. 2, pp. 121–132, 2004.
[76] H. Bourlard and N. Morgan, "Hybrid HMM/ANN systems for speech recognition: Overview and new research directions," in *Adaptive Processing of Sequences and Data Structures*, *Lecture Notes in Artificial Intelligence*. New York: Springer-Verlag, 1998, pp. 389–417.
[77] S. Kruger, M. Schaffner, M. Katz, E. Andelic, and A. Wendemuth, "Speech recognition with support vector machines in a hybrid system," in *Proc. Interspeech*, 2005, pp. 993–996.
[78] J. Platt, "Probabilistic outputs for support vector machines and comparison to regularized likelihood methods," in *Advances in Large Margin Classifiers*, A. Smola, P. Bartlett, B. Schoelkopf, and D. Schuurmans, Eds. Cambridge, MA: MIT Press, 2000, pp. 61–74.
[79] M. Pantic, M. F. Valstar, R. Rademaker, and L. Maat, "Web-based database for facial expression analysis," in *Proc. Int. Conf. Multimedia Expo*, 2005, pp. 317–321.
[80] M. F. Valstar and M. Pantic, "Induced disgust, happiness and surprise: An addition to the MMI facial expression database," in *Proc. Lang. Resour. Eval. Conf.*, 2010, pp. 317–321.
[81] E. Rosenberg, P. Ekman, and J. Blumenthal, "Facial expression and the affective component of cynical hostility in male coronary heart disease patients," *Health Psychol.*, vol. 17, no. 4, pp. 376–380, Jul. 1998.
[82] T. Kirchner, M. Sayette, J. Cohn, R. Moreland, and J. Levine, "Effects of alcohol on group formation among male social drinkers," *J. Stud. Alcohol*, vol. 67, no. 5, pp. 785–794, Sep. 2006.
[83] J. Cohen, "A coefficient of agreement for nominal scales," *Educ. Psychol. Meas.*, vol. 20, no. 1, pp. 37–46, Apr. 1960.
[84] R. Feris, J. Gemmell, K. Toyama, and V. Kruger, "Hierarchical wavelet networks for facial feature localization," in *Proc. IEEE Int. Conf. Autom. Face Gesture Recog.*, 2002, pp. 118–123.
[85] P. Ekman, "Darwin, deception, and facial expression," *Ann. New York Acad. Sci.*, vol. 1000, pp. 105–221, 2003.
[86] P. Yang, Q. Liu, and D. N. Metaxas, "Boosting encoded dynamic features for facial expression recognition," *Pattern Recognit. Lett.*, vol. 30, no. 2, pp. 132–139, Jan. 2009.
[87] J. Whitehill and C. Omlin, "Haar features for FACS AU recognition," in *Proc. 7th Int. Conf. FGR*, Apr. 2006, pp. 97–101.

Michel F. Valstar (M'09) received the M.S. degree in electrical engineering from Delft University of Technology, Delft, The Netherlands, in 2005, and the Ph.D. degree in computer science from Imperial College London, London, U.K., in 2008.

He is a Research Associate in the intelligent Behaviour Understanding Group (iBUG), Department of Computing, Imperial College London. Currently, he is working in the fields of computer vision and pattern recognition, where his main interest is in automatic recognition of human behavior, specializing in the analysis of facial expressions. In 2011, he was the main organizer of the first facial expression recognition challenge, the Facial Expression Recognition and Analysis Challenge 2011, and the organizer of the first audiovisual emotion recognition challenge, the Audio-Visual Recognition Challenge 2011. He has published technical papers at authoritative conferences, including Computer Vision and Pattern Recognition conference, International Conference on Computer Vision, and Transactions on Systems, Man, and Cybernetics—Part B, and his work has received popular press coverage in New Scientist and on BBC Radio. He is also a reviewer for many journals in the field, including Transactions on Affective Computing, Systems, Man and Cybernetics-B and the Image and Vision Computing journal.

Dr. Valstar was the recipient of the British Computing Society British Machine Intelligence Prize for part of his Ph.D. work, in 2007.

Maja Pantic (M'98–SM'06) received the M.Sc. and Ph.D. degrees from Delft University of Technology in 1997 and 2001, respectively.

She is currently a Professor in Affective and Behavioural Computing with the Department of Computing, Imperial College London, London, U.K., and with the Department of Computer Science, University of Twente, Enschede, The Netherlands. She currently serves as the Editor in Chief of the Image and Vision Computing Journal and as an Associate Editor for both the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART B and the IEEE TRANSACTIONS ON MULTIMEDIA.

Prof. Pantic was the recipient of various awards for her work on automatic analysis of human behavior, including the European Research Council Starting Grant Fellowship 2008 and the Roger Needham Award 2011.