14th European Signal Processing Conference (EUSIPCO 2006), Florence, Italy, September 4-8, 2006, copyright by EURASIP

FACIAL EXPRESSION SYNTHESIS THROUGH FACIAL EXPRESSIONS STATISTICAL ANALYSIS

Stelios Krinidis and Ioannis Pitas

Aristotle University of Thessaloniki
Department of Informatics
Box 451, 54124 Thessaloniki, Greece
email: pitas@zeus.csd.auth.gr, stelios.krinidis@mycosmos.gr

ABSTRACT

This paper presents a method for generalizing human facial expressions or personalizing (cloning) them from one person to completely different persons, by means of a statistical analysis of human facial expressions coming from various persons. The data used for the statistical analysis are obtained by tracking a generic facial wireframe model in video sequences depicting the formation of the different human facial expressions, starting from a neutral state. Wireframe node tracking is performed by a pyramidal variant of the well-known Kanade-Lucas-Tomasi (KLT) tracker. The loss of tracked features is handled through a model deformation procedure increasing the robustness of the tracking algorithm. The dynamic facial expression output model is MPEG-4 compliant. The method has been tested on a variety of sequences with very good results, including a database of video sequences representing human faces changing from the neutral state to the one that represents a fully formed human facial expression.

1. INTRODUCTION

Facial analysis and synthesis have become two very important goals for human-centered interface applications. Facial analysis refers to the extraction of information concerning head location, pose and facial feature movement, notably movement of the eyes and mouth from video sequences [1]. Facial synthesis refers to the reverse process of animating a facial model using a set of high-level parameters that control facial pose, expression and gaze [2]. Facial analysis would be useful for several applications, such as eye-tracking, facial expression recognition and visual speech understanding, whereas facial synthesis would be useful for animating virtual characters or digital actors [3]. Together, facial analysis and synthesis *in tandem* would be useful for model-based coding applications, such as video email and video-teleconferencing, as well as for the personalization (cloning) and of representative human facial expressions [4].

The problem of facial expression personalization (cloning) is defined as the transfer of facial expressions from an existing face model to a new model while preserving the emotional feelings embedded in the expressions [5]. The original solution in [5] employed morphing to apply the deformed motion vectors of the source model to the target model. Later, Pyun et al. [6] proposed a new method based on scattered data interpolation [6] and demonstrated its effectiveness in terms of quality and efficiency. In the approach of Pyun et al. [6], the animator provides target key-models, each of which corresponds to a source key-model. Given two or more sets of face key-models, each set representing a different aspect of facial expressions, their approach uses all combinations of key-models from different sets.

Our approach was motivated by the lack of a facial expression personalization system that is able to perform and exploit statistical analysis of the dynamic human facial expressions, i.e., of their formation from the neutral state to the fully expressive one. The goal of this work is to present an integrated system that personalizes human facial expressions through a statistical analysis. Statistical analysis of the facial expressions offers the opportunity to personalize expressions on an individual, with respect to nationality and social class, instead of just transferring a person's expression to another person. Furthermore, in our approach, the aforementioned process is a dynamic one and takes into account the entire video sequence of each facial expression from the neutral state to the fully expressive one. It is not restricted to a single video frame of the expression of interest. All video frames are used and combined to achieve a better result. The statistical analysis is based on the use of displacement vectors. The idea of using displacement vectors was inspired by the Facial Animation Parameters (FAPs) of the MPEG-4 standard. Finally, the portability of the results is achieved through their casting in an MPEG-4 format. Hence, they can be used in any MPEG-4 application.

Before proceeding to the analysis of our method, we should provide some necessary definitions. More specifically, a *dynamic facial expression model* is the set of frame facial expression models (grids) that describe the formation of a facial expression through time. Accordingly, the term *frame facial expression model* corresponds to a grid that is single instance of the dynamic model on a particular video frame.

The main novelty of this paper is the statistical analysis performed on the dynamic facial expression models coming from the video sequences of each of the six human facial expressions that depict each expression from the neutral state to the fully expressive one. Facial grid displacement vectors, inspired by the MPEG-4 standard, were used to perform the statistical analysis, thus avoiding the direct registration of the data (human facial wireframe models). Another novelty of this paper is the presentation and implementation of a MPEG-4 compliant system that is able to perform and exploit statistical analysis (facial grid location and dispersion estimation) of the dynamic human facial expressions, in order to synthesize personalized human facial expressions in two ways, either by applying expressions obtained from one person to the face model of any other person, or by applying representative expressions (extracted from the statistical analysis) to a target face model.

The remainder of the paper is organized as follows. In Section 2, the method of fitting the wireframe facial model to a facial image is presented. Section 3 introduces the tracking algorithm [7] exploiting a physics-based deformation method to compensate for lost features. The statistical facial expression analysis of the data is introduced in Section 4. Finally, experimental results are illustrated in Section 6, while conclusions are drawn in Section 7.

2. WIREFRAME-BASED MODEL FITTING

In this Section, our goal is focused on fitting a facial wireframe model to a face image in a video frame. It is performed in an semi-automatic way for attaining speed, reliability and robustness of the fitting procedure. The facial wireframe model that is used throughout this paper, is the well-known Candide wireframe model

This work has been partially supported by the project Pythagoras II "Efficient techniques for information organization, browsing and retrieval in multimedia", funded by the Greek Ministry of Education and the European Union.

[8]. Candide is a parameterized face mask specifically developed for model-based coding of human faces. The Candide model is superset of the MPEG-4 facial model pattern. The MPEG-4 facial features that are not comprised in Candide model (e.g. ear), are ignored, since they are of no particular significance for facial expression recognition.

The fitting procedure consists of the following steps. First, the facial model is randomly initialized on the face image. The model is assumed to be in its neutral state. As soon as the model is initialized, a number of point correspondences are manually selected, i.e., model nodes are manually matched against facial features in the actual face image. The model nodes of greater significance are chosen to be matched. It has been empirically determined that 5-8 correspondences are enough for a good model fitting. These correspondences are used as the driving power which deforms the rest of the model and matches its nodes against face image points. The facial model is assumed to be a deformable 2-D mesh model. The facial model elements (springs) are assumed to have stiffness k. The driving forces, needed to deform the model, are determined based on the point correspondences between the facial model nodes and the face image features. Each force is defined to be proportional to the Euclidean distance between the model nodes and their corresponding matched feature points on the face image.

Subsequently, the deformation algorithm deforms the facial model and translates the model nodes close to their true position on the face image. As previously mentioned, only a small number (about 5-8 pairs) of point correspondences, is enough to fit the model fairly well to a face image. If the scale difference between the used facial model and the face represented in the image is large, then the number of the required pairs of point correspondences increases to compensate for scale changes. For example, it has been experimentally found that if the model size is 1.5 times larger or smaller than that of the face image, 15 pair point correspondences are needed in order to produce good deformations. This problem can be solved by a preprocessing scaling procedure that scales the model to the size of the face image.

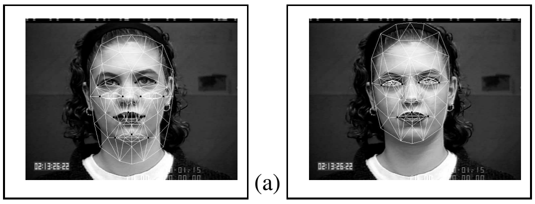

Figure 1: *An example of mesh model fitting on a face image exploiting deformations.*

An example of the facial model fitting procedure on a face image is illustrated in Figure 1. Model fitting was performed using only 7 correspondences between model nodes and face image features. Figure 1a depicts the correspondences between the model nodes and image features, while Figure 1b illustrates the model after deformation. As can be clearly seen in Figure 1, the model nodes were fitted to image features with high accuracy, using few initial correspondences and exploiting the deformation process to accurately match the remaining model nodes to their corresponding image features. In case of any outlying nodes after deformations, their position can be manually changed to achieve the best possible model fitting.

3. MODEL BASED TRACKING

In this Section, an algorithm is presented that is used for tracking the facial feature points of the wireframe model of interest in a video sequence. The algorithm is based on tracking a large number of previously selected feature points in the facial region. Although, features points can be selected automatically, in our case, the tracked feature points are the output of the previous process (Section 2), i.e., they are the nodes of the fitted face model. The test video sequences depict an initially neutral human face, which gradually deforms to produce a particular facial expression. The result of the tracking algorithm is the position of the facial model nodes at intermediate video frames.

Feature points are tracked using a pyramidal implementation of the well-known Kanade-Lucas-Tomasi (KLT) algorithm [7]. A modification of this algorithm that uses a pyramidal representation of the images of interest, is adopted in this paper [7].

As soon as the tracking algorithm computes the displacement of all the tracked features (i.e., the model nodes), the resulting configuration (containing the new positions of the model nodes) is deformed. All model nodes are feature points. The displacements of model nodes, that have not been lost (after tracking), are assumed to be the driving forces of the model deformation, thereby providing an accurate and robust model based facial feature tracking method. This solves a major problem of feature-based tracking algorithms, the gradual elimination of features points with respect to time. In the modified tracking algorithm, the incorporation of the deformation step enables the tracking of features that would have been lost otherwise. Furthermore, feature displacements are enforced to have a uniform distribution considering the features with extreme displacement as lost and their displacement is handled by the deformation procedure. This part of the modified tracking algorithm is used due to that fact that a lot of features are located on plain skin. The tracking algorithm provides a *dynamic facial expression model* for each video sequence, which consists of a series of frame facial expression models, one for each video frame.

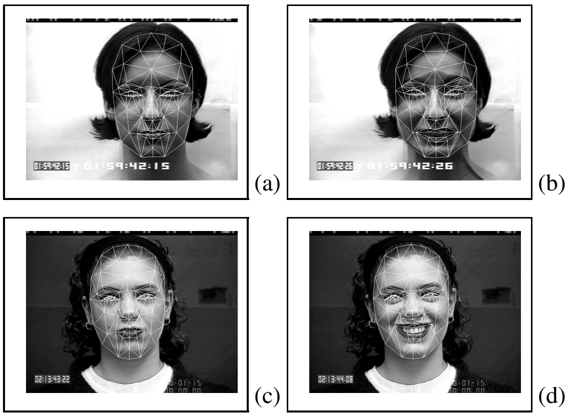

Figure 2: *Model-based facial feature tracking on various video sequences, corresponding to different facial expressions. The left column depicts the first frame of the video sequences, while the right column corresponds to the last frame.*

The proposed tracking algorithm has been applied to various video sequences, as illustrated in Figure 2. The test video sequences comprise of more than 10 frames and demonstrate a gradual change from a neutral state to a particular fully formed facial expression (i.e., laughter, anger, etc.). Figures 2a and 2c depict the model fitted to the neutral face image (i.e., the first frame of the video sequence), while Figures 2b and 2d illustrate the result of model tracking at a fully formed facial expression (last frame of the video sequence).

Furthermore, we have performed an error analysis of the tracking method used by our algorithm. For this experiment we used video sequences containing 24 frames. The feature points (Candide model nodes) of interest were manually tracked for a number of video sequences. The error used in this experiment is defined as:

$$\varepsilon = \frac{1}{N}\sum_{i=1}^{N}\sqrt{(x_i^m - x_i)^2 + (y_i^m - y_i)^2}, \tag{1}$$

where N is the number of tracked feature points, $\mathbf{x}^m = [x^m, y^m]^T$ is the location of a manually tracked feature point (ground truth),

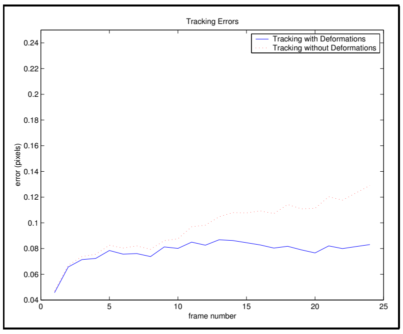

Figure 3: *Node position errors produced by the tracking procedure in video sequences comprising of* 24 *frames.*

while $\mathbf{x} = [x, y]^T$ is the location of the correspondent automatically tracked feature point. Figure 3 illustrates the errors performed by the tracking method used in our algorithm as well as the error occurred by the same tracking method without deforming the tracked feature points after each tracking iteration. It is clearly depicted that both errors are very low, proving the robustness of the tracking algorithm used. Furthermore, as it is obvious in Figure 3, the tracking method exploiting deformations after each iteration produces lower tracking errors. Furthermore, the tracking without model deformation tends to produce error accumulation over time that can possibly cause a collapse of the entire tracking procedure.

4. STATISTICAL ANALYSIS OF FACIAL EXPRESSION GRIDS

In this Section, the statistical analysis of facial expression grids is exploited in order to derive a representative dynamic facial expression model for a set of different dynamic facial expression models corresponding to the formation of a specific facial expression on different persons. The dynamic models of interest are firstly normalized and subsequently subjected to an outlier rejection process. Finally, the representative dynamic facial expression models are extracted and their dispersion is studied.

4.1 Dynamic Facial Expression Model Normalization

A three-step normalization procedure is performed on the set of acquired dynamic facial expression models (grids), so that all dynamic models consist of the same number of frame models and have the same orientation position and size. The facial expression video sequences, and, as a result, the extracted dynamic facial expression models, do not necessarily have the same number of frames. It has been empirically determined that five frames are sufficient to describe an expression without loss of accuracy. Thus, each dynamic facial expression model is constrained to contain only five, uniformly sampled, frame facial expression models, one per frame.

The second step of the normalization procedure is to normalize the frame facial expression models with respect to orientation. This normalization is necessary, among others, for FAP file extraction. We assume that our videos have only frontal pose. More specifically, model rotation around the z-axis is performed in frame facial expression models normalization, so that all frame models are vertical [1].

The last step of the normalization procedure is to normalize the frame facial expression models with respect to size, so that all models have the same FAPUs. Hence, they are scaled in such a way that their facial animation parameter units (FAPUs) are equal. We enforce them to be equal in all the frame models of all the dynamic facial expression models under examination.

4.2 Definition of Facial Expression Model Distances

As soon as all dynamic facial expression models are normalized, their statistical analysis can be performed. The aim of this study is to find a representative dynamic facial expression model for a set of dynamic facial expression models corresponding to different persons posing a specific facial expression. A direct model analysis would require geometric model registration, which may suffer from certain problems. For example, it is difficult to distinguish between scale changes due to camera position or due to the actual size of the head of a person. Instead, an indirect method is applied, which is based on the fact that dynamic facial expression is an "incremental" deformation of the face from the neutral state to the fully expressive one. Therefore, the displacements of the facial features from one video frame to the next one carry sufficient information to characterize a dynamic facial expression model. More specifically, model registration is by-passed by using the model node *displacement vectors*:

$$\tau_m^i(k) = \overrightarrow{M_m^i(k) M_{m+1}^i(k)}, \tag{2}$$

where M^i is the dynamic facial expression model under examination, M_m^i is the m-th frame model of the dynamic model M^i and $M_m^i(k)$ denotes the k-th node of the model M_m^i. The idea of using the displacement vectors is inspired by the MPEG-4 standard [9] and FAP definition there in.

Let us, now, define the distance d_v between two displacement vectors $\tau_m^i(k)$ and $\tau_m^j(k)$:

$$d_v(\tau_m^i(k), \tau_m^j(k)) \triangleq \|\tau_m^i(k) - \tau_m^j(k)\|, \tag{3}$$

where $\|\cdot\|$ denotes either the Euclidean distance. Furthermore, the distance d_f between two frame facial expression models M_m^i and M_m^j is defined as:

$$d_f(M_m^i, M_m^j) \triangleq \frac{1}{N} \sum_{k=1}^{N} d_v(\tau_m^i(k), \tau_m^j(k)), \tag{4}$$

where N is the number of facial model nodes. On the same basis, the distance d between two dynamic facial expression models M^i and M^j is defined as:

$$d(M^i, M^j) = \frac{1}{N'-1} \sum_{m=1}^{N'-1} d_f(M_m^i, M_m^j), \tag{5}$$

where $N'-1$ is the number of the displacement vectors sets corresponding to the N' frame facial expression models of each normalized dynamic model.

4.3 Outlying Facial Expression Model Rejection

Before proceeding to the estimation of the location and dispersion of the facial expression models, an outlier rejection process has to be employed to remove outlying facial expression models. Let us assume that we have N'' dynamic facial expression models, one for each facial expression sequence. Facial model distance definitions can be used for outlying facial expression model rejection, before estimating their location and dispersion. The chosen outlier trimming process is performed as follows [10]. A distance $d_t(M^i)$ of one dynamic facial expression model M^i from the rest of the models that belong in the same class is defined. Models of each class are sorted according to $d_t(M^i)$ and $a\%$ of them that possess the largest of $d_t(M^i)$ are trimmed away, thereby freeing the model data from possible outliers. The trimming process is applied at the frame facial expression model level, i.e., $\alpha\%$ of the models that are located too far away from the rest are removed from the sets that are formed by the frame facial expression models which correspond to the same video frame. The distance between a frame facial expression model M_m^i and the rest frame models corresponding to the video frame m, is defined as:

$$d_{tf}(M_m^i) = \frac{1}{N''-1} \sum_{\substack{j=1 \\ j \neq i}}^{N''} d_f(M_m^i, M_m^j). \tag{6}$$

4.4 Estimation of Location and Dispersion of Dynamic Facial Expression Models

Once the facial expression model data have been freed from outliers, we can proceed to the estimation of the location and dispersion of the facial expression model data. Location estimation is essentially the estimation of a representative dynamic facial expression model [11] out of a set of dynamic facial expression models, e.g. of the six basic facial human expressions as expressed by various humans. The same procedure can be applied to find the representative dynamic facial expression of one particular expression (e.g. smile) for one person in case we have multiple video sequences of the same person while smiling. This can be done by finding the *generalized median dynamic facial expression model* [11], which is defined as the dynamic model $M_{med} \in \mathbf{U}$ that minimizes the sum of distances to all dynamic models belonging to $\mathbf{M}$, i.e.,

$$M_{med} = \arg\min_{M\in\mathbf{U}} \sum_{M^i\in\mathbf{M}} d(M, M^i), \qquad (7)$$

where $\mathbf{U}$ is the domain of all possible dynamic facial expression models, while $\mathbf{M}$ is the set of all dynamic facial expression models of a particular class (e.g. dynamic smiles of many persons, different smile sequences of the same person). The generalized median of graphs has been investigated and found useful in the field of statistical graph analysis. The definition $d(M, M^i)$ depends on the distance metric used. It can be found using a greedy search algorithm [11].

Dynamic model dispersion can be measured using a variant of the Median of the Absolute Deviations (MAD) estimator [10] modified to fit the dynamic facial expression models:

$$\hat{\sigma}_{MAD} \triangleq \text{med}\left\{d(M^1, M_{med}), \ldots, d(M^{N''}, M_{med})\right\}, \qquad (8)$$

where M_{med} is the generalized median dynamic facial expression model [10]. The above mentioned dispersion estimation metrics indicate the dispersion, i.e. the dissimilarity of the various facial expression, e.g. among different persons.

A flow chart of the overall algorithm is illustrated in Figure 4. At the end of the entire procedure, MPEG-4 FAP files describing the representative expressions and their progressive formation with respect to time, are extracted.

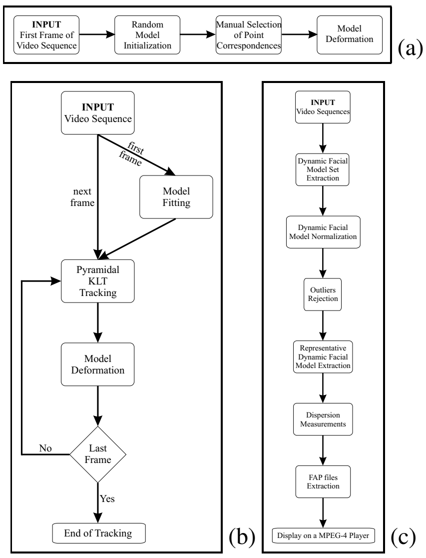

Figure 4: ***Flow diagrams**. **(a)** Model fitting algorithm on a face image, **(b)** Facial feature tracking algorithm, and **(c)** Overall algorithm.*

5. FACIAL EXPRESSION SYNTHESIS/PERSONALIZATION

In this Section, the facial expression synthesis and personalization is presented. The goal is to synthesize personalized human facial expressions in a number of ways. First, we can clone facial expressions, i.e. apply dynamic facial expressions obtained for one person to another person (target face), e.g. we can make a person smile as a well known actor does. We can also apply the extracted representative (generalized median) dynamic facial expressions to a particular person (e.g. we can make a person smile in a similar way as a particular social/ethnic group does).

Dynamic facial expression synthesis can be performed in the following way. First of all, the wireframe model used is fitted to the target face image following the procedure described in Section 2. This initialization step is necessary in order to obtain the geometry of the target face. Afterwards, if the aim is to produce a personalized expression of the first type (i.e., to make a person express like another person), we apply the displacement vectors of the source dynamic model to the target one, in such a way that a new dynamic model similar to the source one is produced. On the same basis, if we want to produce a personalized expression of the second type (i.e., to make a person express in a representative way), we move the corresponding nodes of the target model accordingly to the extracted FAP file of the representative dynamic facial expression model. The personalized model could acquire the proper texture by a well-known texture mapping technique [12].

6. EXPERIMENTAL RESULTS

To evaluate our method, we use the well-known Cohn-Kanade expression database [13] as input in our experiments. The database contains image sequences of over 200 subjects in the age range of 18 to 50 years. 69% of them are females, while 31% are males. 81% of the database subjects are Euro-Americans, 13% are Afro-Americans, and 6% belong to other racial groups. The motivation for the selection of this database originates from its content, i.e., the fact that image sequences depict the formation of human facial expressions from the neutral state to the fully expressive one.

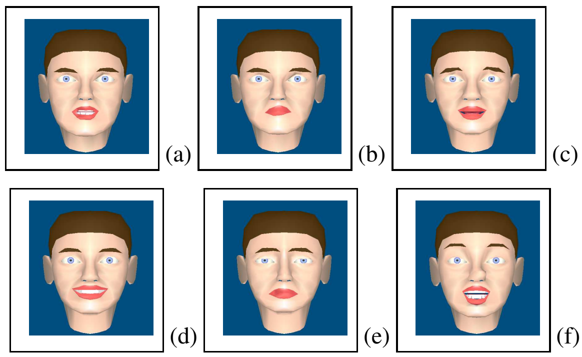

Figure 5: *The representative dynamic facial expression model corresponding to **(a)** anger, **(b)** disgust, **(c)** fear, **(d)** laughter, **(e)** sadness, and **(f)** surprise.*

Table 1: ***Dispersion** $\hat{\sigma}_{MAD}$ of the dynamic facial expression models for the six basic human facial expressions.*

Anger	Disgust	Fear	Laughter	Sadness	Surprise
11,13	9,80	11,39	10,89	13,46	13,57

The first set of experiments shows the efficiency of the proposed algorithm with respect to the extraction of representative dynamic facial expression models for each of the six basic human facial expressions. Table 1 shows the dispersion of the dynamic models of the six basic human facial expressions. Dispersion measure used correlates quite well. Laughter and disgust models possess the least dispersion, while surprise models show the largest dispersion among individuals in the Cohn-Kanade expression database [13]. This can be explained by the fact that surprise contains stronger node displacements. It is worth noticing that trimming reduces class dispersion, as expected. The representative (generalized median) dynamic facial expression models have also been converted to MPEG-4 FAP files, so that they can be displayed by any MPEG-4 player. In our case, the facial model developed in the context of the European project ACTS MoMuSys was used to this end. Figure 5a shows two frames (frame 3 and 5) of the representative dynamic facial expression model for anger, while Figures 5b, 5c, 5d, 5e and 5f illustrate the representative dynamic models for disgust, fear, laughter, sadness and surprise, respectively, obtained using the generalized dynamic facial expression median model after trimming at the frame model level. It has been found experimentally that this approach produces the best subjective facial expression results. This is best noticed when observing the entire MPEG-4 face animation sequence.

(a) (b) (c)

Figure 6: ***Facial expression cloning***. ***(a)*** *original facial expression (smile),* ***(b)*** *and* ***(c)*** *cloned facial expression (smile).*

(a) (b) (c)

Figure 7: *Example of mapping a representative expression (sadness) to a new target person.* ***(a)*** *Original expression,* ***(b)*** *synthesized expression using models extracted by our dataset, and* ***(c)*** *representative sadness expression.*

Finally, the introduced method is tested with respect to personalization, i.e., mapping personal facial expressions from one actor to another or mapping representative expressions to a new target person. Figure 6 illustrates the expressions from one person 6a cloned to two different persons 6b and 6c as described in Section 5. We can notice that the synthetic facial expression results are fairly natural. Again, this is better understood when visualizing the entire synthetic facial animations. Furthermore, Figure 7 depicts the personalization of a representative expression (sadness) applied to a new target person. Figure 7a illustrates the original expression of the person, while 7b shows the synthesization of the person's own expression (the models used are extracted from the original expressions of the same person shown in the first row). Figure 7c depicts the corresponding expression obtained when applying the representative dynamic facial expression model to the same person. Again, it can be seen that the synthetic facial expression results are fairly natural.

7. CONCLUSION

In this paper we presented a complete method for the statistical analysis of human facial expressions and for adapting human facial expressions of one subject to new target faces, thus obtaining personalization. The analysis was performed in the framework of facial expression analysis and synthesis and produces results that are MPEG-4 compatible. The data used for the statistical analysis were obtained by tracking a generic facial wireframe model in video sequences depicting the formation of different human facial expressions, starting from a neutral state to a fully expressive one, using a pyramidal variant of the well-known Kanade-Lucas-Tomasi (KLT) tracker. Any loss of tracked features is handled through a physics-based deformation stage, after each single tracking step, providing accuracy and reliability. Tracking initialization is performed in a semi-automatic fashion. The facial wireframe model is fitted to a neutral facial image using a deformable shape modeling approach which is robust, fast and accurate. The output is MPEG-4 compliant and can be utilized in any MPEG-4 player. The method has been tested on a variety of sequences with very good results, including the Cohn-Kanade database of video sequences representing human facial expressions. It has been shown that the proposed method performs a number of intermediate steps in a reliable and accurate way and thus achieves a good statistical analysis of facial expressions and a realistic facial expression personalization.

REFERENCES

[1] A. Nikolaidis and I. Pitas, "Facial feature extraction and pose determination," *Pattern Recognition, Elsevier*, vol. 33, no. 11, pp. 1783–1791, November 2000.

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

[3] Z. Ruttkay, J. Hendrix, P. Hagen, A. Lelievre, H. Noot, and B. Ruiter, "A facial repertoire for avatars," in *Proceedings of the Workshop Interacting Agents*, Enschede, The Netherlands, 2000.

[4] S. Krinidis, I. Buciu, and I. Pitas, "Facial expressions analysis and synthesis: A survey," in *International Conference on Human Computer Interaction (HCI'03)*, Crete, Greece, June 2003.

[5] J. Y. Noh and U. Neumann, "Expression cloning," in *Proceedings of SIGGRAPH (SIGGRAPH'01)*, 2001, pp. 277–288.

[6] H. Pyun, Y. Kin, W. Chae, H. Y. Kang, and S. Y. Shin, "An example-based approach for facial expression cloning," in *Proceedings of ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 2003, pp. 167–176.

[7] J. Y. Bouguet, "Pyramidal implementation of the lucas kanade feature tracker," Intel Corporation, Microprocessor Research Labs, OpenCV Documents, Tech. Rep., 1999.

[8] M. Rydfalk, "CANDIDE: A parameterized face," Ph.D. dissertation, Linkoping University, Linkoping, 1978.

[9] MPEG, "Coding of audio-visual objects - Part 2: Systems," *ISO/IEC 14496-1:2001*, 2nd Edition 2001.

[10] I. Pitas and A. N. Venetsanopoulos, *Nonlinear Digital Filters: Principles and Applications*. Norwell, MA: Kluwer Academic Publishers, January 1990.

[11] X. Jiang, A. Munger, and H. Bunke, "On median graphs: Properties, algorithms and applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 19, pp. 1144–1151, October 2001.

[12] D. Blythe, B. Grantham, M. J. Kilgard, T. McReynolds, and S. R. Nelson, "Advanced graphics programming techniques using OpenGL," in *SIGGRAPH'99 Course Notes*, August 1999.

[13] T. Kanade, J. Cohn, and Y. Tian, "Comprehensive database for facial expression analysis," in *Proceedings of International Conference on Face and Gesture Recognition*, March 2000, pp. 46–53.