Journal of information and communication convergence engineering
http://jicce.org

J. Inf. Commun. Converg. Eng. 11(3): 207-215, Sep. 2013 **Regular paper**

# 3D Facial Landmark Tracking and Facial Expression Recognition

**Gérard Medioni, Jongmoo Choi*, Matthieu Labeau, Jatuporn Toy Leksut, and Lingchao Meng, *Member, KIICE***

Computer Vision Lab, Institute for Robotics and Intelligent Systems, University of Southern California, Los Angeles, CA 90089, USA

## Abstract

In this paper, we address the challenging computer vision problem of obtaining a reliable facial expression analysis from a naturally interacting person. We propose a system that combines a 3D generic face model, 3D head tracking, and 2D tracker to track facial landmarks and recognize expressions. First, we extract facial landmarks from a neutral frontal face, and then we deform a 3D generic face to fit the input face. Next, we use our real-time 3D head tracking module to track a person's head in 3D and predict facial landmark positions in 2D using the projection from the updated 3D face model. Finally, we use tracked 2D landmarks to update the 3D landmarks. This integrated tracking loop enables efficient tracking of the non-rigid parts of a face in the presence of large 3D head motion. We conducted experiments for facial expression recognition using both frame-based and sequence-based approaches. Our method provides a 75.9% recognition rate in 8 subjects with 7 key expressions. Our approach provides a considerable step forward toward new applications including human-computer interactions, behavioral science, robotics, and game applications.

**Index Terms**: Computer vision, Facial expression recognition, Facial landmark tracking, 3D-face tracking

## I. INTRODUCTION

Facial expressions are a fundamental element of our daily social interactions. Faces exhibit a rich set of details about someone's mental status, intentions, concerns, reactions, and feelings [1]. Expressions and other facial gestures are an essential component of nonverbal communication. They are critical in emotional and social behavior analysis, humanoid robots, facial animation, and perceptual interfaces.

We aim to develop a prototype system that takes real-time video input from a webcam, tracks facial landmarks from a subject looking at the camera, and provides expression recognition results. The system includes use of a 3D generic face model [2, 3], adaptation of the generic face model to a user, tracking of 3D facial landmarks, analysis of the set of expressions, and real-time frame rate implementation. The approach is illustrated in Fig. 1.

First, we use a real-time 3D head tracking module, which was developed in our lab [4], to track a person's head in 3D (6 degrees of freedom). We use a RGB video as the input, detect frontal faces [5, 6], extract facial landmarks from a neutral face [7-10], deform a 3D generic face model to fit the input face [3, 4, 10], and track the 3D head motion from the video using the updated 3D face.

Second, the main contribution is a landmark tracking algorithm. We combine 2D landmark tracking and 3D face pose tracking. In each frame, we predict the locations of 2D facial landmarks using the 3D model, check the consistency between 2D tracking and prediction, and update the 3D landmarks. This integrated tracking loop enables efficiently

**Received** 03 April 2013, **Revised** 29 April 2013, **Accepted** 10 May 2013
***Corresponding Author** Jongmoo Choi (E-mail: jongmooc@usc.edu, Tel: +1-213-740-0991)
Computer Vision Lab, Institute for Robotics and Intelligent Systems, University of Southern California, Los Angeles, CA 90089, USA.

Open Access http://dx.doi.org/10.6109/jicce.2013.11.3.207 print ISSN: 2234-8255 online ISSN: 2234-8883

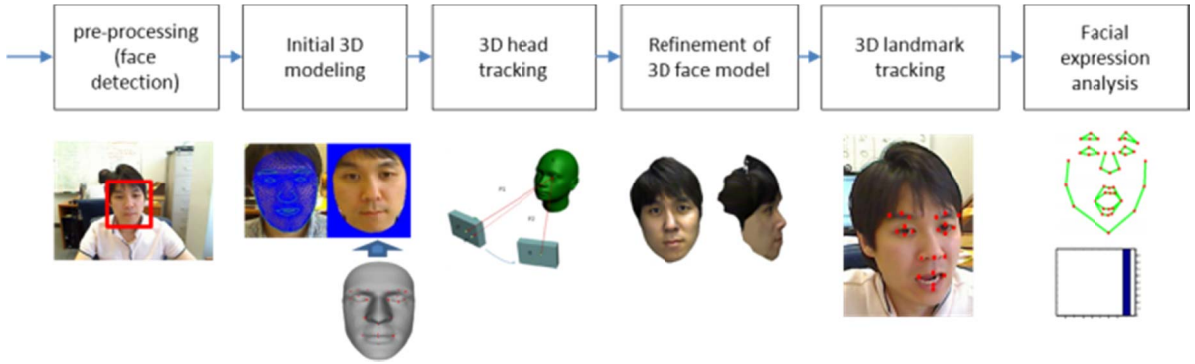


**Fig. 1.** Overview of the system.

tracking the deformations of the non-rigid parts of a face in the presence of large 3D head motion.

Third, we have conducted experiments for facial expression recognition using a standard dynamic time warping algorithm. Our method provides a 75.9% recognition rate (8 subjects) with 7 key expressions (joy, surprise, fear, anger, disgust, sadness, and neutral).

The rest of the paper is organized as follows. Section II provides a short review of the related work. Section III describes the proposed 3D facial landmark tracking, our approach to expression inference and the improvements needed, and preliminary experimental results using facial expression databases. Conclusions are given in Section IV.

## II. RELATED WORK

Active shape models (ASM) [7, 8, 11] approximate 2D shape deformation as a linear combination of basis shapes which are learned using Principal Component Analysis [12]. An active appearance model (AAM) [8, 13] learns not only shapes but also appearance models from texture information. 3D deformable models have also been proposed. A combined 2D+3D AAM is presented in [13]. In [14], Blanz and Vetter show a 3D morphable model for facial animation and face recognition. In [15], Gu and Kanade present a 3D deformable model consisting of sparse 3D points and patches associated with each point. Constrained local models, per-forming an exhaustive local search for each landmark around a current estimate and finding the global non-rigid shape parameters, are presented in [16].

Facial expression analysis in the presence of wide pose variations is important for interaction with multiple persons. In [17], Zhu and Ji proposed a normalized single value decomposition to estimate the pose and expression simultaneously. Vogler et al. [18] uses ASM to track reliable features and a 3D deformable model to infer the face shape and pose from tracked features. Taheri et al. [19] shows that the affine shape-space, an approximation to the projective shape-space, for 2D facial landmark configurations has Grassmannian properties and non-rigid deformations can be represented as points on the Grassmannian manifold, which can be used to perform expression analysis without the need for pose normalization. Facial expression analysis is performed on tracked facial features [17, 19, 20], which can be represented in low-dimensional manifolds [21].

Our method leverages the accurate estimation of a 3D pose and surface model to infer non-rigid motion of 3D facial features. In addition, intra/interclass variations in the low-dimensional spatio-temporal manifold are handled by a novel spatio-temporal alignment algorithm to recognize facial expressions.

## III. SYSTEM MODEL

### A. Overview

The goal of the system is, given as input a 2D video (taken from a webcam) of a face, to recognize in real-time the emotions expressed by the person. In order to perform the recognition, we need to collect appearance and shape information and classify it into expressions. This information relies on facial motion and deformations. We consequently track a set of feature points, called *facial landmarks*, that we define using the eyes, the eyebrows, the nose, and the mouth. Indeed, through a determined number of these particular points, we can analyze the deformations and accurately determine facial expressions: they carry important information, as it is easy for a human to identify another human's expressions.

The approach starts with face detection. Then, we need to locate predefined key points on the face and to track them along the video stream. It is a difficult task; first, for external reasons, changes in resolution, illumination, and occlusions occur regularly. The subject will probably move; the difficulty is not translations, but about rotations of the head (change of pose)—these change the appearance of the

face. The next difficulty is the nature of the face; it is deformable and highly complex. We can differentiate between non-rigid landmarks, that are localized on deformable parts of the face and so much more difficult to track, and the rigid ones. These two categories require different kinds of tracking.

As shown in Fig. 1, our system is divided in three main modules: 3D face tracking, landmark tracking, and expression recognition. 3D face tracking, developed by our group, is described in [4]. It detects the face in a 2D image and fits a generic 3D model to it using ASM: 3D landmarks of the generic model are aligned with the 2D landmarks of the face found using ASM, allowing a warping of the generic 3D mesh to obtain a 3D model of the person.

To perform landmark tracking, we then have two sources of information: the 2D images from the webcam and the 3D face model that is simultaneously tracked. The idea here is to use a classic tracking algorithm to track the 2D landmarks, whose initial positions are obtained from the reconstructed 3D model and estimated 3D pose of the model. We can then monitor this 2D tracking using the tracked 3D model.

The difficulty here concerns non-rigid landmarks: their positions cannot be checked by the 3D model (which is rigid) and we need to rely only on the 2D image to track these deformations. We have proposed and implemented a tracking loop that uses the information of the 3D model, a set of 3D landmarks corresponding to the 2D landmarks in facial images, to evaluate the 2D tracking, and the results of 2D tracking to update the 3D model (i.e., 3D landmarks). It can handle face deformation, using projection of the tracked points on an "authorized" area, determined by the natural constraints of the face (as a point of the inferior lip can only move vertically in a specific range in the 3D face reference frame).

## B. 3D Facial Landmark Tracking

### 1) 3D Face Modeling and Tracking Using a Webcam

Described in [4], our approach consists of face detection, initial 3D model fitting, 3D head tracking and re-acquisition, and 3D face model refinement. We need a 3D model, which can either be a generic model, or a specific one retrieved from a database. In the initial 3D modeling step, the model is warped orthogonally to the focal axis in order to fit to the user's face in an input image, by matching 2D facial landmarks extracted at runtime. Our tracker uses this 3D model to compute 3D head motion despite partial occlusions and expression changes, even in case of erroneous corresponding points. The robustness is achieved by acquiring new 2D and 3D keypoints along the tracking, for instance coming from a profile view. At each iteration, only the most relevant keypoints, according to the camera field of view, are matched between the 3D model rendering and the input video.

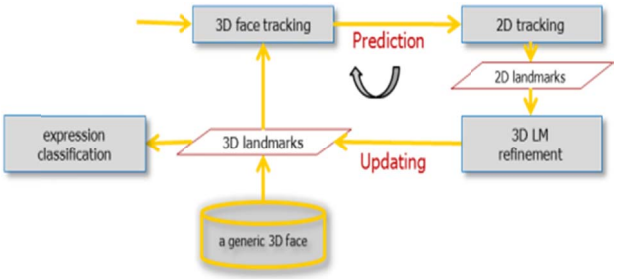


**Fig. 2.** Key idea of 3D landmark tracking.

In addition, we have a recovery mechanism in case the tracker loses track. Tracking failures are identified by a sharp decrease in the number of tracked features, and a background process matches features in the next frame(s) against the reference frontal face. Refinement of the 3D face model, a background process, improves the accuracy of 3D face tracking and 3D landmark tracking.

### 2) 3D Landmark Tracking

#### (a) Key idea

Our method consists of prediction of 2D landmarks and an update of 3D landmarks (See Fig. 2). At each iteration, we predict the 2D locations of all 3D landmarks using the estimated head motion. Then, we update some of the 3D landmarks if and only if our prediction does not explain the observed 2D points that are tracked by a 2D tracking algorithm from the previous frame.

The prediction process is done by 3D pose estimation. Given a 3D point ($X$), a 2D point can be found by the projection $x = PX$, where $P$ is the projective projection matrix and $(x, X)$ is the pair of 2D and 3D points in the homogeneous coordinate system. Note that our 3D head tracker gives the projection matrix at each frame.

In the comparison step, we compute the distance between a predicted location and a tracked location by a 2D feature tracker. If there are no expression changes or tracking errors, the locations should be the same. However, there are several sources of error, such as a 3D head tracking error, 2D landmark tracking error, and facial deformation.

We validate the quality of 3D face tracking by comparing the predicted and tracked locations from a set of rigid points. For instance, points on the nose should not be deformed and if the distances between projected nose points and tracked nose points are large, we can update the global 3D head motion by minimizing the re-projection error.

We represent the error between our prediction and observations as

$$\Delta(i) = \left\| f(P)g\big(M(i)\big) - m'(i) \right\|, \qquad (1)$$

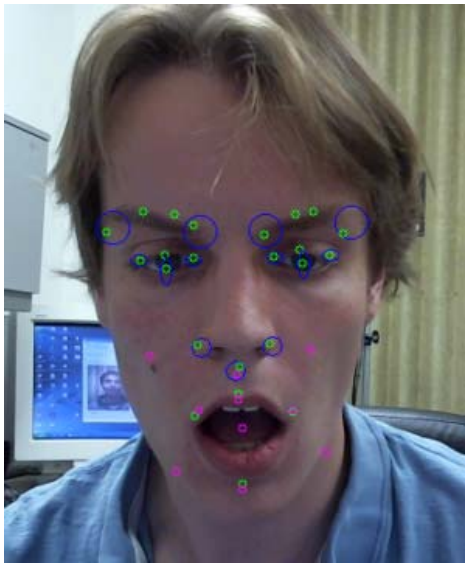

**Fig. 3.** Visualization of the deformation step.

where $f(P)$ represents the projection matrix including the estimation error, $g(M(i))$ represents a deformed location of a 3D point $M(i)$, and $m'(i)$ is a tracked 2D point. If the difference $\Delta(i)$ is small enough, we can skip the current frame and process the next input frame. Otherwise, if the distance is large, we try to minimize the distance by updating rigid motion $f(P)$. If we cannot minimize the error by finding a new rigid motion, it is likely that there is deformation of the 3D landmark. In this case, we search the non-rigid transformation $g(M(i))$. In our approach, we use a set of geometrical constraints which bound the locations of 3D landmarks in 3D space. For instance, the top point of the upper lip should be located in the vertical center line between the two end points of the mouth in 3D space. We define such a constraint for each 3D landmark and use it to correct the wrong projection of tracked landmarks.

#### *(b) Implementation of 3D landmark tracking*

We describe the actual implementation of our 3D landmark tracking loop. Fig. 3 shows the pipeline for each frame. Using the camera parameters for the frame $t$, $P(t) = K[R,T]$ is retrieved by the head tracking framework. The 3D landmarks $X(i,t-1)$ from frame $t$–1 are projected to obtain the set $x(i,t) = P(t)X(i,t-1)$.

We use the Lucas-Kanade tracker (LKT) [22] to track our landmarks $m(i,t-1)$ to frame $t$ and obtain $m(i,t)$. Then, we check the validity of these 2D landmarks, using the 3D landmarks from the previous frame. We try to determine if there is any tracking error due to LKT or the deformation of the face, and if so, we correct it. We also update the 3D landmarks on the face model to keep track of the deformations from one frame to the next.

**Fig. 4.** Landmark tracking with 3D head motion.

We define several areas that have their own deformations (i.e., mouth, eyes, and eyebrows). What is important to the consistency of our tracking for an area is the distance

$$D_{area}(x(t), m(t)) = \sum_{i<area} ||x(i,t) - m(i,t)||. \quad (2)$$

If this distance goes above a certain threshold, we go into the deformation step: the consistency of the tracked points $m(i,t)$ is not evaluated using their distance to $x(i,t)$, but they are projected on a convex space defined by 2 or 3 points, depending on the area.

Fig. 3 shows the deformation step. In purple are the points defining the convex spaces on which the tracked points are projected, while the green points are the tracked points after the deformation step. Most of the authorized movements are defined by segments (upper and lower lips, eyelids, and eyebrows) but the corner points of the lips are projected into a triangle.

The points defining the convex spaces are taken from the 3D model, and we do a reverse projection of the tracked points $m(i,t)$ to obtain their corresponding 3D coordinates $M(i,t)$. The new 3D coordinates $M'(i,t)$ are projected back on the image as $m'(i,t)$. In a same way, we evaluate our tracking's consistency with the distance

$$D'_{area}(x(t), m(t)) = \sum_{i<area} ||x(i,t) - m'(i,t)||. \quad (3)$$

If the distance goes above a (larger) threshold, we use a stronger tracker for re-acquisition. However, in practice, the need for re-acquisition almost never occurs. Indeed, in case of motion, the deformation step helps the LKT by limiting potential drifting, and significantly reduces the impact of the aperture problem.

### *C. Facial Expression Recognition*

Since human facial expressions are dynamic in nature, we focus on expression recognition on a sequence of images. Fig. 5 shows the overview of our approach. Given a video input,

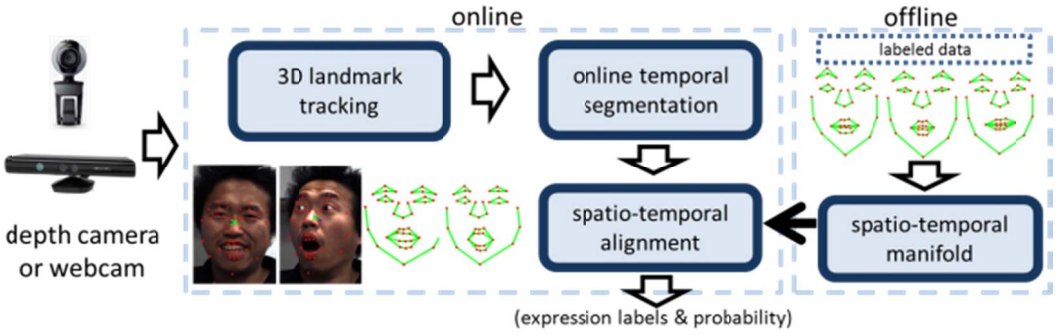


**Fig. 5.** Overview of facial expression recognition.

we perform online temporal segmentation and compute the distance between input data and stored labeled data using spatio-temporal alignment algorithms. In this paper, we focus on the spatio-temporal alignment. In the following sections, we describe the details of our approach.

### *1) Sequence-Based and Frame-Based Recognition*

Understanding the dynamics of expression changes is important. We can classify expression recognition methods into two categories: sequence-based and frame-based recognition methods. A sequence-based method uses a segmented sequence as an input to the system, while a frame-based method takes only a single frame. The segmentation is by a temporal segmentation algorithm or a sliding window technique. In general, a sequence-based method can provide more accurate results than a frame-based one since the neighbor frames are highly correlated and a sequence-based method utilizes a data fusion technique. However, sequence-based methods might have significantly delayed responses.

### *2) Recognition Using Sequences*

We compute the distance between $N$ observations from input video, $X = \{x(1), x(2), \dots, x(N)\}$, and $M$ observations from stored data, $Y = \{y(1), y(2), \dots, y(M)\}$, where each element of the sequence $y(j)$ contains extracted facial landmarks. Note that the length of sequences $(N, M)$ need not be the same $(N \neq M)$. We assume that an input sequence is segmented from streaming data and our input sequence contains a transition of facial expressions from "neutral" to a specific expression (e.g., "joy").

Since the absolute scale of a shape is independent of facial expressions, we normalize each shape vector as $X_{normalized} = \frac{X}{||X||}$ so all of the shape vectors lay on a unit hypersphere (Fig. 6).

Each observation, containing a set of shapes, is a sequence (a trajectory) on the sphere. The issue is that if we compare "joy" and "surprise", the end points that depict the mouth opening might be similar to each other. Hence, we want to compare the entire sequences instead of comparing them only at the end points which correspond to static shapes (i.e., two frames or two set of landmarks).

The dynamic time warping (DTW) algorithm is a well-known method that aligns two data sets containing sequential observations [23, 24]. We align two sequences using dynamic programming and a distance function (we use the cosine distance between two landmarks). We then compute the Chebyshev distance between the two aligned sequences.

The Chebyshev distance between two points $P = (x(1), x(2), \dots, x(n))$ and $Q = (y(1), y(2), \dots, y(n))$ is defined as:

$$D(P, Q) \coloneqq \max_i |x(i) - y(i)|. \quad (4)$$

This equals the limit of the Minkowski distance and hence it is also known as the L-∞ norm. In our facial expression recognition, each aligned element represents the correlation value between signals. Hence, our distance is represented as

$$D(X, Y) = \frac{1}{\min_{\{C(x'(1),y'(1)),C(x'(2),y'(2)),\dots,C(x'(L),y'(L))\}}}, \quad (5)$$

where $C(\cdot)$ is the correlation function between aligned data $(x'(j), y'(j))$ and $L$ is the minimum number of data points $(L = \min(N, M))$. This concept is illustrated in Fig. 6.

### *3) Evaluation of Facial Expression Recognition*

We present two evaluation results of facial expression recognition. First, we present results of a frame-based method using the k-nearest neighbor (K-NN) algorithm. Second, we show results of a sequence-based algorithm using the DTW algorithm.

#### *(a) Data and setup*

We defined 7 expressions: joy, surprise, fear, anger, disgust, sadness, and neutral. We collected 182 (13 subjects × 7 expressions × 2 sessions) videos from 13 people (14 videos

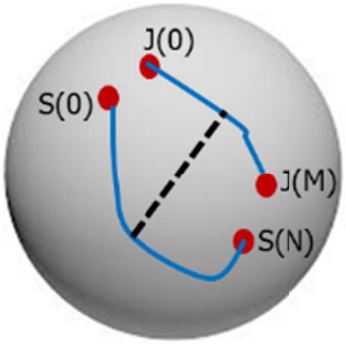


**Fig. 6.** The L∞ norm (dashed line) between two aligned sequences: even though the shapes between two starting (J(0), S(0)) and between two ending points (J(M), S(N)) can be similar to each other, the L∞ distance discriminates between the two sequences.

per subject). To perform expressions, we showed subjects typical image search results from Google using corresponding keywords (e.g., "joy face", "fear face", "surprise face", …). Each person was asked to sit in front of a webcam (0.8–1.2 m) for a short period (about 10 seconds for each expression) in an office environment. We controlled the distance from the subject to the camera so that the distance from the eye to the camera was in a range between 150–200 pixels. We collected two videos for each expression from each subject: session 1 and session 2. Videos from sessions 1 and 2 were used for training and testing, respectively. The image resolution was VGA (640 × 480) and we recorded videos at 15 fps.

***(b) Experiment 1: using a single frame***

We collected a subset of data containing 6 people's videos: 42 videos from each session. From each video, we used 170 frames for testing. The gallery set contains 7,140 frames (170 frames × 7 expressions × 6 subjects) and the probe set has the same number of frames.

All of the landmarks are aligned to a single reference shape, which is the first shape from the first subject in the database. To align the two sets of 2D points, we compute the affine transformation using the homography relationship $q = Hp$, where $H$ is a 3×3 matrix and $p$ and $q$ are 2D points in the homogeneous coordinate system.

We compute the distance between each frame ($X$) in the probe set against the gallery set $\{Y(1), Y(2), \dots, Y(N)\}$. To classify an input, we use the K-NN algorithm with an L2 distance measure (K = 24). To identify the class of an input data, we select one of the gallery data points which has the maximum score

$$id\ of\ X = \arg\max_j \{dist(X, Y(j))\},\ Y(j) < G. \quad (6)$$

Table 1 shows a confusion matrix. The average recognition rate is 55.2%.

**Table 1.** Frame-based recognition rate (%) using k-nearest neighbor algorithm (average = 55.2%)

| Actual class | Predict class | | | | | | |
|---|---|---|---|---|---|---|---|
| | **Neutral** | **Joy** | **Fear** | **Surprise** | **Anger** | **Disgust** | **Sadness** |
| Neutral | 86.19 | 0 | 13.10 | 0.48 | 0.24 | 0 | 0 |
| Joy | 0 | 96.19 | 0.48 | 3.33 | 0 | 0 | 0 |
| Fear | 36.90 | 0.00 | 36.43 | 0.71 | 4.05 | 0 | 21.90 |
| Surprise | 17.38 | 24.52 | 0 | 42.62 | 0 | 0 | 15.48 |
| Anger | 4.29 | 0 | 12.14 | 0 | 66.90 | 16.67 | 0 |
| Disgust | 26.19 | 4.76 | 7.62 | 0 | 36.67 | 23.57 | 1.19 |
| Sadness | 5.95 | 0 | 11.19 | 0 | 22.38 | 25.95 | 34.52 |

**Table 2.** Sequence-based recognition rate (%) using dynamic time warping algorithm (average = 75.8%)

| Actual class | Predict class | | | | | | |
|---|---|---|---|---|---|---|---|
| | **Neutral** | **Joy** | **Fear** | **Surprise** | **Anger** | **Disgust** | **Sadness** |
| Neutral | **75** | 0 | 0 | 0 | 0 | 0 | 25 |
| Joy | 0 | **100** | 0 | 0 | 0 | 0 | 0 |
| Fear | 0 | 0 | **100** | 0 | 0 | 0 | 0 |
| Surprise | 0 | 25 | 0 | **50** | 0 | 0 | 25 |
| Anger | 0 | 0 | 0 | 0 | **100** | 0 | 0 |
| Disgust | 0 | 17 | 0 | 0 | 0 | **50** | 30 |
| Sadness | 33 | 0 | 0 | 0 | 0 | 0 | **67** |

***(c) Experiment 2: using a sequence***

We used a subset of data for experiment 2. We built gallery datasets from all of the first videos (session 1) from 8 people's data. The first gallery database included 29 tracked sequences from the subjects:

$$G = \{g(1,1), g(2,1), \dots, g(7,1), g(1,2), \dots, g(Eg, Ng)\},$$

where $Eg$ is the number of expressions and $Ng$ is the number of subjects. We built a probe dataset from all of the second videos (29 tracked sequences from session 2):

$$P = \{p(1,1), p(2,1), \dots, p(7,1), p(1,2), \dots, p(Ep, Np)\},$$

where $Ep$ is the number of expressions and $Np$ is the number of subjects. We compute the distance between

$$g(i,j) = X = \{x(1), x(2), \dots, x(N)\} \text{ and}$$
$$p(k,m) = Y = \{y(1), y(2), \dots, y(M)\},$$

where each element of the sequence $y(j)$ contains extracted facial landmarks. Note that the length of sequences $(N, M)$ need not be the same. Given a pair of data $(X, Y)$, we align them using the DTW algorithm and compute a score between two aligned sequences as

$$score(X,Y) = \min\{C(x'(1), y'(1)), C(x'(2), y'(2)), \dots, C(x'(L), y'(L))\}, \quad (7)$$

where $C(\cdot)$ is the correlation function between aligned data $(x'(j), y'(j))$, and $L = \min(N, M)$ is the minimum number of data. To identify the class of an input data point, we select one of the gallery data point which has the maximum score:

$$id\ of\ X = \arg\max_j \{score(X, Y(j))\},\ Y(j) < G. \quad (8)$$

Table 2 shows a confusion matrix and the average recognition rate is 75.8% for the 7 expressions.

## IV. CONCLUSIONS

We present a system that is able to, from a low-resolution video stream, detect the face of a subject, build a 3D model of it, and track it in real-time, as well as obtain predefined facial feature points, the facial landmarks. Both rigid motion and non-rigid deformation are tracked, within a large range of head movements. In addition, we developed face recognition algorithms using a single frame and a sequence of images. We have validated our approach with a real database containing 7 facial expressions (joy, surprise, fear, anger, disgust, sadness, and neutral).

Further work will consist of improving the accuracy of both deformation tacking and face recognition algorithms. Updating the 3D face model should enable providing more accurate rigid and non-rigid tracking. Analysis of facial expression manifolds is one of the key tasks that needs to be investigated to improve the recognition rate.

## REFERENCES


[ 1 ] J. A. Russell, "Emotion, core affect, and psychological construction," *Cognition & Emotion*, vol. 23, no. 7, pp. 1259-1283, 2009.

[ 2 ] W. K. Liao, D. Fidaleo, and G. Medioni, "Robust: real-time 3D face tracking from a monocular view," *EURASIP Journal on Image and Video Processing*, vol. 2010, article no. 5, 2010.

[ 3 ] J. Choi, G. Medioni, Y. Lin, L. Silva, O. Regina, M. Pamplona, and T. C. Faltemier, "3D face reconstruction using a single or multiple views," in *Proceedings of the 20th International Conference on Pattern Recognition*, Istanbul, Turkey, pp. 3959-3962, 2010.

[ 4 ] J. Choi, Y. Dumortier, S. I. Choi, M. B. Ahmad, and G. Medioni, "Real-time 3-D face tracking and modeling from a webcam," in *Proceedings of the IEEE Workshop on Applications of Computer Vision*, Breckenridge, CO, pp. 33-40, 2012.

[ 5 ] P. Viola and M. J. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 7, no. 2, pp. 137-154, 2004.

[ 6 ] OpenCV: Open Source Computer Vision [Internet], Available: http://opencv.org/.

[ 7 ] T. F. Cootes and C. J. Taylor, "A mixture model for representing shape variation," in *Proceedings of the 8th British Machine Vision Conference*, Essex, UK, 1997.

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

[ 9 ] S. Milborrow and F. Nicolls, "Locating facial features with an extended active shape model," in *Proceedings of the 10th European Conference on Computer Vision*, Marseille, France, pp. 504-513, 2008.

[10] G. Medioni, J. Choi, C. H. Kuo, and D. Fidaleo, "Identifying noncooperative subjects at a distance using face images and inferred three-dimensional face models," *IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans*, vol. 39, no. 1, pp. 12-24, 2009.

[11] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham, "Active shape models: their training and application," *Computer Vision and Image Understanding*, vol. 61, no. 1, pp. 38-59, 1995.

[12] K. Pearson, "On lines and planes of closest fit to systems of points in space," *Philosophical Magazine*, vol. 2, no. 6, pp. 559-572, 1901.

[13] J. Xiao, S. Baker, I. Matthews, and T. Kanade, "Real-time combined 2D+3D active appearance models," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Washington, DC, pp. 535-542, 2004.

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

[15] L. Gu and T. Kanade, "3D alignment of face in a single image," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, New York, NY, pp. 1305-1312, 2006.

[16] D. Cristinacce and T. F. Cootes, "Feature detection and tracking with constrained local models," in *Proceedings of the 17th British Machine Vision Conference*, Edinburgh, UK, pp. 929-938, 2006.

[17] Z. Zhu and Q. Ji, "Robust real-time face pose and facial expression recovery," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, New York, NY, pp. 681-688, 2006.

[18] C. Vogler, Z. Li, A. Kanaujia, S. Goldenstein, and D. Metaxas, "The best of both worlds: combining 3D deformable models with active shape models," in *Proceedings of the 11th IEEE International Conference on Computer Vision*, Rio de Janeiro, Brazil, 2007.

[19] S. Taheri, P. Turaga, and R. Chellappa, "Towards view-invariant expression analysis using analytic shape manifolds," in *Proceedings of the IEEE International Conference on Automatic Face & Gesture Recognition and Workshops*, Santa Barbara, CA, pp. 306-313, 2011.

[20] W. K. Liao and G. Medioni, "3D face tracking and expression inference from a 2D sequence using manifold learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Anchorage, AK, 2008.

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

[22] B. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision (DARPA)," in *Proceedings of the DARPA Image Understanding Workshop*, Washington, DC, pp. 121-130, 1981.

[23] C. S. Myers and L. R. Rabiner, "A comparative study of several dynamic time-warping algorithms for connected word recognition," *Bell System Technical Journal*, vol. 60, no. 7, pp. 1389-1409, 1981.

[24] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 26, no. 1, pp. 43-49, 1978.

**Gérard Medioni**
received the Diplome d'Ingenieur from ENST, Paris in 1977, abd a M.S. and Ph.D. from the University of Southern California in 1980 and 1983, respectively. He has been at USC since then, and is currently a Professor of Computer Science and Electrical Engineering, co-director of the Institute for Robotics and Intelligent Systems (IRIS), and co-director of the USC Games Institute. He served as Chairman of the Computer Science Department from 2001 to 2007. Professor Medioni has made significant contributions to the field of computer vision. His research covers a broad spectrum of the field, such as edge detection, stereo and motion analysis, shape inference and description, and system integration. He has published 4 books, over 50 journal papers and 150 conference articles, and is the recipient of 8 international patents. Prof. Medioni is on the advisory board of the IEEE Transactions on PAMI journal, associate editor of the International Journal of Computer Vision, associate editor of the Pattern Recognition and Image Analysis journal, and associate editor of the International Journal of Image and Video Processing.

**Jongmoo Choi**
received the B.S. degree in Physics, the M.S. degree in Cognitive Science, and the Ph.D. degree in Computer Engineering from Sungkyunkwan University, Korea. From 1999 to 2003, he was the director of the research center of Dream Mirh Co. Ltd., where he developed face recognition algorithms and SDKs for applications such as machine readable travel documents and biometric entrance management systems. The technology has been transferred to Korea Identification Inc. and is being employed in many important places, such as Incheon International Airport and the Korea National Policy Office. From 2004 to 2006, he was a research assistant professor at the Intelligent Systems Research Center, Sungkyunkwan University, Korea and he invented 3D robotic sensors based on signal separation coding. He is currently a senior research associate at the Institute for Robotics and Intelligent Systems, University of Southern California, USA. The technologies developed at USC have been transferred to Primesense, OpenNI, HP, Samsung, IQ Engines, Skycomp, and Northrop Grumman Corporation. Dr. Choi's current research interests include 3D face modeling, 3D face recognition, 3D face tracking, facial expression recognition, robust parameter estimation, and event/activity analysis in video.

**Matthieu Labeau**
received the M.S. degree in Artificial Intelligence from Katholieke Universiteit Leuven, Leuven. He was a visiting scholar at the Institute for Robotics and Intelligent Systems, University of Southern California, Los Angeles, CA.

**Jatuporn Toy Leksut**
was born in Thailand in 1988. She has received the Royal Thai Government Scholarship to support her studies in the US since 2007. She received the B.S. degree in computer science from George Washington University, Washington, DC in 2012. She is currently a Ph.D. student in computer science at the Institute for Robotics and Intelligent Systems, University of Southern California, Los Angeles, CA. Her research interests include computer vision and computer graphics.

**Lingchao Meng**
is a graduate student at the University of Southern California (Aug 2011–May 2013). He received his master's degree from Ming Hsieh Department of Electrical Engineering at USC. In the first year of his master's degree, Lingchao continued to explore his field of interest: intelligent robotics. At the beginning of the second academic year, Lingchao learned that computer vision plays an important role in many key applications for robots such as controlling, navigation, detecting events, modeling objects, interaction, and automatic inspection. Then he tried his best to obtain knowledge and experience about computer vision and is dedicated to applying vision techniques to robotics.