GRAPHICAL MODEL BASED FACIAL FEATURE POINT TRACKING IN A VEHICLE ENVIRONMENT

Serhan Coşar, Müjdat Çetin, Aytül Erçil

Sabancı University
Faculty of Engineering and Natural Sciences
Orhanlı- Tuzla, 34956 İstanbul, TURKEY
serhancosar@su.sabanciuniv.edu,{mcetin,aytulercil}@sabanciuniv.edu

ABSTRACT

Facial feature point tracking is a research area that can be used in human-computer interaction (HCI), facial expression analysis, fatigue detection, etc. In this paper, a statistical method for facial feature point tracking is proposed. Feature point tracking is a challenging topic in case of uncertain data because of noise and/or occlusions. With this motivation, a graphical model that incorporates not only temporal information about feature point movements, but also information about the spatial relationships between such points is built. Based on this model, an algorithm that achieves feature point tracking through a video observation sequence is implemented. The proposed method is applied on 2D gray scale real video sequences taken in a vehicle environment and the superiority of this approach over existing techniques is demonstrated.

1. INTRODUCTION

Facial feature point tracking is an important step in problems such as video based facial expression analysis and human-computer interaction. Generally, a facial expression analysis system consists of three components: feature detection, feature tracking, and expression recognition. Feature detection involves detecting some distinguishable points that can define the movement of facial components. This may involve, detection of eyes, eye brows, mouth or feature points of these components. Then comes the tracking part which consists of tracking the detected feature points. Finally, according to tracking results of these feature points, the expression recognition component outputs results such as happy, sad, surprised, etc. This paper is an application of facial feature point tracking to video sequences taken in a vehicle environment [1]. The output of the proposed work can be considered as an input to a facial expression analysis system in a vehicle for driver fatigue detection and driver behavior modeling. Such a fatigue detection system in vehicles is very important because statistics show that transporting by vehicle is quite unsafe compared to the other transporting options. Police reports say that driver faults are the main reason of more than 80% of the accidents. Cooperating the emerging elements of digital life and technology with the recent automobile technology will produce intelligent vehicles that can warn the driver automatically in possible accident situations.

For feature point tracking, roughly there are two classes of methods in literature: general purpose approaches, face-specific approaches. One of the general purpose approaches is moving-point-correspondence methods [2, 3]. The smooth motion, limited speed, and no occlusion assumptions make these methods inapplicable to facial-feature tracking. Another approach is the patch correlation method [4, 5] which is sensitive to illumination and object-pose variations. There are also some optical-flow based methods [6, 7] which often assume image-intensity constancy for corresponding pixels, which may not be the case for facial features.

Compared with the general-purpose feature-tracking techniques, the face-specific methods are more effective. There are methods which use Gabor Filters [8, 9] to track facial feature points. Also, active appearance (AAM) or active shape models (ASM) [10, 11, 12] are used to track feature points based on a face model. Additionally, the work in [8] tracks feature points based on spatial and temporal connections using non-parametric methods.

Generally, feature point tracking is done by using a temporal model that is based on pixel values. Consequently, these methods are sensitive to illumination and pose changes, and ignore the spatial relationships between feature points. This affects the tracking performance adversely, causes drifts and physically unreasonable results when the data are noisy or uncertain due to occlusions. In [13], a method where the spatial relationships are taken into account is proposed for contour tracking. However, since the method is based on non-parametric estimation techniques, it is rather computationally intensive.

In this paper feature point tracking is performed in a framework that incorporates the temporal and spatial information between feature points. This framework is based on graphical models that have recently been used in many computer vision problems. The model is based on a parametric model in which the probability densities involved are Gaussian. The parametric nature of the models makes the method computationally efficient. The spatial connections between points allow the tracking to continue reasonably well by exploiting the information from neighboring points, even if a point disappears from the scene or cannot be observed. The feature values from video sequences are based on Gabor filters. Filters are used in a way to detect the edge information in the image, to be sensitive to different poses, orientations and different feature sizes. Tests on videos recorded in a vehicle environment showed that tracking of facial feature points is performed successfully.

2. PROPOSED METHOD

2.1. Preprocessing

Gabor filters are used as a preprocessing stage for the observation section of the proposed work. The filters are selected as in [14]. Then as in [8], consecutive frames are convolved with 24 filters consisting of 6 different orientations and 4 different wavelengths. The magnitude and phase of the complex outputs of the filtering is compared using the similarity metric in [14]. The location of the point in the next frame is found by comparing the filter outputs from the current and the next frame. This is used as the data component of the method.

2.2. Graphical Models

Graphical models can be defined as a marriage of graph theory and probability theory. The visualization property of graph theory makes even a complex model clear and understandable. This provides a powerful, general framework for developing statistical models of computer vision problems.

Generally a graph G is defined by a set of nodes V, and a corresponding set of edges E. The neighborhood of a node $s \in V$ is defined as $N(s) = \{t|(s,t) \in E\}$. The models are divided into two main categories: directed and undirected graphs. Directed graphs are graphs in which there is a causal relation between random variables. In undirected graphs the relation is bidirectional. Some examples of graphical models are illustrated in Figure 1.

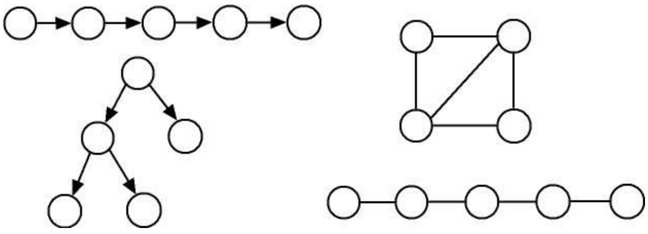


Fig. 1. Some examples of graphical models

Graphical models usually associate each node $s \in V$ with an unobserved, hidden random variable (x_s), and a noisy local observation (y_s). Let $x = \{x_s|s \in V\}$ and $y = \{y_s|s \in V\}$ denote the sets of all hidden and observed variables, respectively. This simply makes the factorization of a joint probability function $p(x,y)$ as shown below.

$$p(x,y) = \frac{1}{Z} \prod_{(s,t)\in E} \psi_{s,t}(x_s,x_t) \prod_{(s)\in V} \psi_s(x_s,y_s) \qquad (1)$$

Here, $\psi_{s,t}(x_s,x_t)$ is the edge potential between hidden variables. The other term, $\psi_s(x_s,y_s)$ is the observation potential.

The graphical model that is used in the proposed method, when we track two feature points, is shown in Figure 2. Each hidden variable (x_s) in the model is a vector with four elements. Assuming the movement of the feature points are in 2D, these four elements are x-coordinates, y-coordinates, velocity at x-axis and velocity at y-axis of the points. The observed nodes (y_s) are a vector with two elements; x - coordinates and y-coordinates of observation data.

In this notation, (x_t^1) means hidden variable of the first feature point at time t and (y_{t+1}^2) is the observed variable of the second feature point at time t+1.

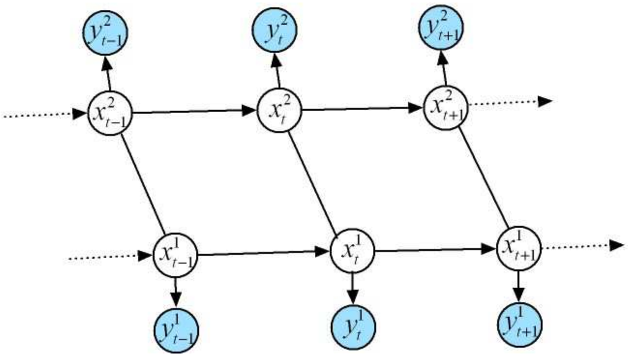


Fig. 2. The graphical model used in this work

The selection of the edge potentials mentioned in this section is explained in 2.3–2.5.

2.3. Temporal Model

The temporal model makes the connection of the feature points with the previous value of the points. This is based on the translation model shown below.

$$x_{t+1} = A \cdot x_t + w \qquad w \backsim N(0, Q) \tag{2}$$

Here, A is the translation matrix. Q is the covariance matrix of the noise which is a normal distribution with zero-mean. Assuming the points move with a constant velocity and the point coordinates and velocities are independent of each other, these are selected as:

$$A = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} Q = \begin{pmatrix} \sigma_x^2 & 0 & 0 & 0 \\ 0 & \sigma_y^2 & 0 & 0 \\ 0 & 0 & \sigma_u^2 & 0 \\ 0 & 0 & 0 & \sigma_v^2 \end{pmatrix} \tag{3}$$

Hence the temporal connection between two nodes involves a Gaussian distribution and the edge potential can be defined as follows:

$$\psi_{t,t+1}(x_t, x_{t+1}) = \\ \alpha \exp\{-\frac{1}{2} \begin{bmatrix} x_{t+1}^T & x_t^T \end{bmatrix} \begin{bmatrix} Q^{-1} & -Q^{-1}A \\ -A^TQ^{-1} & A^TQ^{-1}A \end{bmatrix} \begin{bmatrix} x_{t+1}^T \\ x_t^T \end{bmatrix} \} \tag{4}$$

2.4. Spatial Model

The spatial model defines the spatial connection between feature points. This connection is selected to simply use the expected spatial distance between feature points, for example; the distance between the eye corners. According to these, the spatial connection is selected as below:

$$\psi_{1,2}(x_1, x_2) = \\ \alpha \exp\{\begin{bmatrix} x_1 - (x_2 - \triangle x) \end{bmatrix}^T \Sigma^{-1} \begin{bmatrix} x_1 - (x_2 - \triangle x) \end{bmatrix}\} \tag{5}$$

Here, $\triangle x$ is a four elements vector containing only the distance on x–axis as: $\triangle x = \begin{bmatrix} \triangle & 0 & 0 & 0 \end{bmatrix}^T$. The covariance matrix is selected as below:

$$\Sigma = \begin{pmatrix} \sigma_x'^2 & 0 & 0 & 0 \\ 0 & \sigma_y'^2 & 0 & 0 \\ 0 & 0 & \sigma_u'^2 & 0 \\ 0 & 0 & 0 & \sigma_v'^2 \end{pmatrix} \tag{6}$$

2.5. Observation Model

The extraction of the observations from video sequences is explained in section 2.1.

The observation model makes the connection between the hidden random variable x_s and the noisy local observation variable y_s. The model is as follows:

$$y_t = C \cdot x_t + v \qquad v \backsim N(0, R) \tag{7}$$

Here, C is the observation matrix and R is the covariance matrix of the noise which is a normal distribution with zero-mean. These are selected as follows:

$$C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} R = \begin{pmatrix} \sigma_x''^2 & 0 \\ 0 & \sigma_y''^2 \end{pmatrix} \tag{8}$$

As a result, the relation between x_s and y_s is a normal distribution and it is defined as follows:

$$\psi_s(x_s, y_s) = N(C \cdot x_s, R) = p(y_s|x_s) \tag{9}$$

2.6. Loopy Belief Propagation Algorithm

In many computer vision and image processing applications, the main target is to find the conditional density function $p(x_s|y)$. For graphs which are acyclic or tree–structured, the desired conditional distributions can be directly calculated by a local message–passing algorithm known as *belief propagation* (BP). In chain-structured graphs, this algorithm is equal to Kalman or Particle filtering. For cyclic graphs, Pearl [15] showed that *belief propagation* produces excellent empirical results in many cases. The algorithm is as follows: Each node $t \in V$ calculates a message $m_{t,s}(x_s)$ to be sent to each neighboring node $s \in N(t)$:

$$m_{t,s}(x_s) = \alpha \int_{x_t} \psi_{s,t}(x_s, x_t)\psi_t(x_t, y_t) \times \prod_{u \in N(t) \backslash s} m_{u,t}(x_t) dx_t \tag{10}$$

Each node combines these messages and its own observation and produces its own conditional density function :

$$p(x_s|y) = \alpha \psi_s(x_s, y_s) \prod_{t \in N(s)} m_{t,s}(x_s) \tag{11}$$

Since the relations in the model are selected as Gaussian, the two steps of the algorithm shown above simplify to updating means and covariances. For this reason, it works faster than non-parametric methods. Since the hidden random vector consists of the x,y axis coordinates and velocities, the mean values of the normal distributions are the estimation of these values. Each update step is done using the current and the previous data, as a result the algorithm used becomes a filtering algorithm. For the update equations please see [16].

3. EXPERIMENTAL RESULTS

The performance of the proposed work is shown by using the data recorded in a laboratory environment and in a vehicle environment [1]. The data consist of translation of head and external occlusion of some points. These video sequences consist of real–world head movements, facial movements, gestures etc. So this gives a sense of the practical application of the proposed method.

For a simplicity, only four eye corner points are tracked. The covariance matrices in the temporal, spatial and observation model are selected suitably according to the videos. The results of the proposed work in videos taken in a laboratory environment are shown in Figure 3–b and results from the vehicle data are shown in Figure 4–b. For comparison, the results of an algorithm that exploits only the temporal relations [8] are shown in Figure 3–a and 4–a. Green dots in the result images are the estimated point locations obtained using the observation data up to now.

As shown in Figures 3–b and 4–b, tracking of feature points is successfully done in the case of head translation or external occlusion. On the other hand, as seen in Figure 3–a and 4–a, the method that only use the temporal relation cannot track well and drifts occur and unreasonable tracking results.

The occlusion sequence is recorded by occluding a part of the face by hand. In this case it is assumed that there is an occlusion detector. The data term is closed when there is an occlusion. For a regular comparison, this is applied for both proposed method and the method in [8].

4. CONCLUSION AND FUTURE WORK

In this paper a robust feature point tracker for real time applications, such as driver fatigue detection, driver behavior modeling etc., is developed. The significant advantage of the algorithm is the incorporation of the temporal and spatial information. So if a point disappears from the scene due to an occlusion, the information from the neighboring points will allow the tracking of the lost point to continue successfully. Another advantage of the method is the computational efficiency. The parametric assumptions make computations simpler with respect to non-parametric techniques.
For future work, the model will be improved to track all facial feature points as an input to a facial expression analysis system or a fatigue detection system in vehicles.

5. ACKNOWLEDGEMENTS

This work was partially supported by the European Commission under Grants FP6-2004-ACC-SSA-2 (SPICE) and MIRG-CT-2006-041919, the Turkish State Planning Organization under the DRIVE-SAFE project, the New Energy and Industrial Technology Development Organization (NEDO) of Japan under the project ”International Research Coordination of Driving Behavior Signal Processing based on Large Scale Real World Database”.

6. REFERENCES

[1] H. Abut, H. Erdogan, A. Ercil, B. Curuklu, H.C. Koman, F. Tas, A.O. Argunsah, S. Cosar, B. Akan, H. Karabalkan, E. Cokelek, R. Ficici, V. Sezer, S. Danis, M. Karaca, M. Abbak. M.G. Uzunbas, K. Ertimen, C. Kalaycioglu, M. Imamoglu, C. Karabat, and M. Peyic, “Data collection with ’uyanik’: Too much pain; but gains are coming,” in *Proc. Biennial on DSP for In-Vehicle and Mobile Systems*, Istanbul, Turkey, June 2007.

[2] Chetverikov D and Verestoy J, “Tracking feature points: a new algorithm,” in *Proceedings of the international conference on pattern recognition*, 1998, p. 14361438.

[3] Rangarajan K and Shah M, “Establishing motion correspondence,” *CVGIP: Image Understanding*, vol. 54, pp. 56–73, 1991.

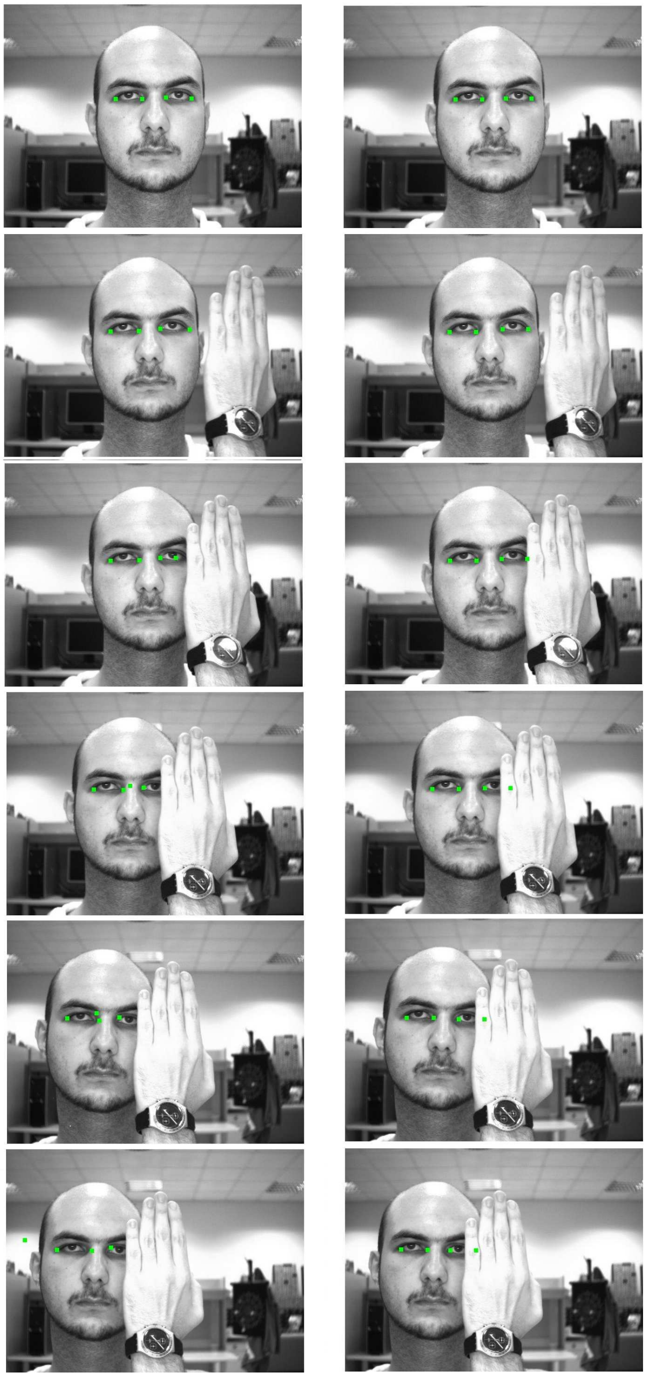
[4] Bretzner L and Lindeberg T, “Feature tracking with automatic selection of spatial scales,” *Computer Vision Image Understanding*, vol. 71, no. 3, pp. 385–392, 1998.

[5] Shapiro L, Wang H, and Brady J, “A matching and tracking strategy for independently-moving, non-rigid objects,” in *Proceedings of the 3rd British machine vision conference*, 1992, pp. 306–315.

[6] Meyer F and Bouthemy P, “Region-based tracking using affine motion models in long image sequences,” *Computer Vision Image Understanding*, vol. 60, pp. 119140, 1994.

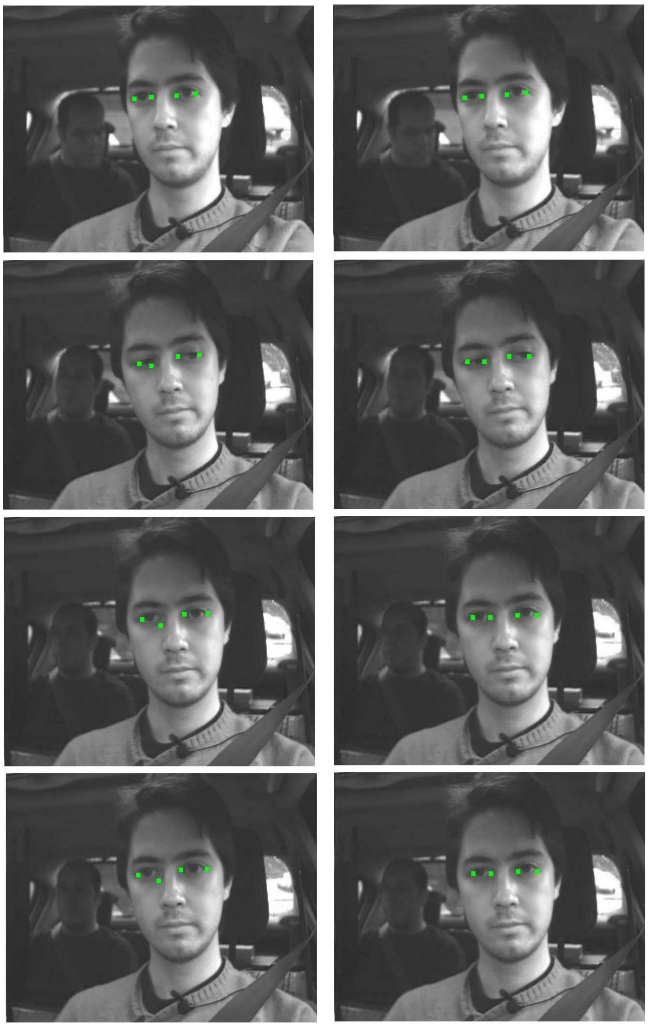
[7] Zheng Q and Chellappa R, “Automatic feature point extraction and tracking in image sequences for arbitrary camera motion,” *International Journal of Computer Vision*, vol. 15, pp. 3176, 1995.

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



(a) (b)

Fig. 3. Ideal environment tracking result of (a) the method in [8] (b) the proposed method



(a) (b)

Fig. 4. Vehicle environment tracking result of (a) the method in [8] (b) proposed method

[9] Ji Q and Yang X, "Real-time eye, gaze, and face pose tracking for monitoring driver vigilance," *Real-Time Imaging*, vol. 8, no. 5, pp. 357–377, 2002.

[10] Cootes T F, Edwards G J, and Taylor C J, "Active appearance models," in *Proceedings of the European conference on computer vision*, 1998, p. 484498.

[11] Xiao J, Baker S, Matthews I, and Kanade T, "Real-time combined 2d+3d active appearance models," *CVPR*, vol. 2, pp. 535–542, 2004.

[12] Wan K, Lam K, and Chong N, "An accurate active shape model for facial feature extraction," *Pattern Recognition Letters*, vol. 26, no. 15, pp. 2409–2423, 2005.

[13] Su C and Huang L, "Spatio-temporal graphical-model-based multiple facial feature tracking," *EURASIP Journal on Applied Signal Processing*, vol. 13, pp. 2091–2100, 2005.

[14] David S. Bolme, “Elastic bunch graph matching,” M.S. thesis, Colorado State University, 2003.

[15] J. Pearl, *Probabilistic Reasoning in Intelligent Systems*, Morgan Kaufman, San Mateo, 1988.

[16] E. Sudderth, “Embedded trees: Estimation of gaussian processes on graphs with cycles,” M.S. thesis, MIT, 2002.