Face Tracking in Video with Hybrid of Lucas-Kanade and Condensation Algorithm

Chong Luo
Department of Computer Science
National University of Singapore
luochong@comp.nus.edu.sg

Tat Seng Chua
Department of Computer Science
National University of Singapore
chuats@comp.nus.edu.sg

Teck Khim Ng
Department of Computer Science
National University of Singapore
ngtk@comp.nus.edu.sg

ABSTRACT

In this paper, we present a robust face tracking system for video indexing and retrieval. Our face tracker is designed based on the condensation algorithm. The strength of our face tracking system is in the incorporation of Lucas-Kanade feature tracker in the measurement stage of condensation. Skin color and facial feature points are used for tracking. The pros and cons of using color and facial feature points complement each other and ensure the effectiveness of our system. We also adopt a bi-directional tracking approach to enhance the robustness. We demonstrate the efficacy of our technique in the challenging task of tracking faces in various video sources.

1. INTRODUCTION

Face tracking is a subject of great interest in recent years due to its wide applications in digital video management, surveillance, and human-computer interactions. Most face tracking methods use only visual information, while several methods [11,14] use both visual and acoustic cues. The visual based methods are either color-based [3,8,9], model-based [2,16] or facial feature-based [5,7].

The majority of visual-based face tracking methods consider only color feature [3,8,9], which has been shown to offer several significant advantages over geometric features for certain tasks in visual perception [13]. It has been demonstrated that human skin colors are clustered in the color space and the distribution of human skin colors can be represented by a two-dimensional Gaussian function in the normalized color space [15]. Therefore, skin color can be considered for face tracking. The use of only color cues, however, is susceptible to false alarms especially when there are large skin-like-color regions in the background. Thus, there is a need to incorporate other features for robustness and accuracy.

One approach for robustness is to use model-based techniques [2,16]. This approach employs a deformable face model and defines some parameters representing the head position, facial features and facial expression appearing in the video frames. Most of the model-based methods target at applications like human-computer interactions, in which the face and shoulder regions occupy a large portion of the frame [2,16]. These methods, however, are hard to be adopted for video indexing task where the size, location and appearance of faces are not known a priori. Moreover, the size of the faces is normally too small for detailed shape analysis.

Another approach for robustness in face tracking is to track the specific feature points in the face regions [5,7]. The tracking of these facial features is precise and reliable in the case of planar movement. It is however very sensitive to out of plane rotation, occlusion, scale or resolution changes. Once the tracker loses the feature points, it is hard to regain proper tracking.

Other approaches based on stochastic theories such as Kalman filtering are also used [12]. Kalman filtering is however inadequate in video indexing tasks because it has the uni-model Gaussian assumption and therefore cannot represent simultaneous alternative hypotheses.

Our face tracking system is designed to improve upon the existing tracking algorithms. We adopt the particle filtering, or condensation framework. The condensation framework has the ability to represent simultaneous alternative hypotheses of face position. Since the tracking is propagated stochastically over time, temporal information is fully utilized for robustness.

In order to use the condensation framework effectively to track faces from frame to frame, the choice of sample set and the design of propagation steps should be established. The strength of our face tracking system is in the incorporation of Lucas-Kanade feature tracker in the measurement stage of condensation. The features we used are skin color and facial feature points. It has been demonstrated that the color cue is able to handle partial occlusion [13], and is tolerant to rotation in depth and scale changes etc. However, using only color cue will not give accurate face location. On the other hand, facial feature point tracking gives accurate location of face but it is not as robust as color cue especially when the pose of face changes. The combined use of color and facial feature points therefore complement each other. Our face tracking system is designed based on this consideration.

This paper is organized as follows: section 2 introduces the condensation algorithm. Section 3 describes our face tracking system. Section 4 presents the experimental results. Finally, section 5 concludes the paper and discusses future work.

2. THE CONDENSATION ALGORITHM

Our face tracking algorithm is based on the condensation framework. The name condensation stands for conditional density propagation. It is based on factored sampling and has been used to iteratively analyze successive images in a video sequence [4]. The algorithm was presented in [4], in which a successful tracking of curves in dense visual clutter was demonstrated. The condensation algorithm keeps multiple hypotheses of the likelihood of the interested object location by a set of weighted samples. The process at each time-step is a self-contained iteration of factored sampling, and is composed of three stages: selection, prediction and measurement.

At time-step t, the input is a weighted sample set from time-step t-1, i.e. $Input_t = \{ s_{t-1}^{(n)}, \pi_{t-1}^{(n)}, n = 1,\ldots N \}$, where $s_{t-1}^{(n)}$ is the n^{th} sample at time-step t-1 with the weight $\pi_{t-1}^{(n)}$. The output of time-step t is $Output_t = \{ s_t^{(n)}, \pi_t^{(n)}, n = 1,\ldots N \}$.

At the selection stage, the task is to select N samples from $\{ s_{t-1}^{(n)} \}$ according to their probabilities $\{ \pi_{t-1}^{(n)} \}$. Some elements, especially those with high probabilities, may be chosen several times leading to identical copies of elements in the new set. Others with relatively low probabilities may not be chosen at all. The selected sample set is denoted as $\{ s_t'^{(n)} \}$.

At the prediction stage, $s_t^{(n)}$ is generated from $s_t'^{(n)}$ using a specific rule. Usually, the rule is defined based on a piecewise linear assumption. Assuming that the movement of $s_t^{(n)}$ is linear, it could be written as:

$$s_t^{(n)} = s_t'^{(n)} + c_t + w \tag{1}$$

where c_t is the displacement from time-step t-1 to t, and w is a Gaussian white noise.

At the measurement stage, the probabilities of all the N samples $\{ \pi_t^{(n)} \}$ are updated based on the evidence given by the features chosen for the sample set. As these will be the probabilities used in stage 1 (i.e. selection stage) of the next iteration, it is therefore important to design good measurement metric and methodology. In our system, we measure skin color and the reliability of Lucas-Kanade tracker for the tracking of facial feature points.

3. OUR FACE TRACKING SYSTEM

Our face tracking system is based on the condensation algorithm and requires the support of an existing face detection method to provide the initial estimation. A generic skin color model is built in advance from a large number of faces collected from different video streams. The entire system is composed of four parts as shown in Figure 1.

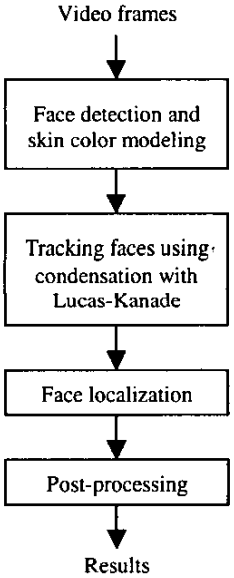

Figure 1: System overview

In the first step, we employ a well-established frontal face detector [1] to detect faces on every other I-frames. Possible faces are searched at every possible positions and scales. Once a face is detected, the frame that contains it is considered a seed frame. The pixels in the detected face regions are also used to train a specific skin color model.

In the second step, the tracking is conducted forward and backward from every seed frame until it meets another seed frame or a video boundary. We employ the condensation algorithm to identify locations with high probability of occurrence of faces. The skin color probability evaluates the likelihood of the samples falling within a face region. In parallel, some distinct facial features are selected in every seed frame and they are tracked using the Lucas-Kanade method. The position of these tracked features and their matching error offers a complementary measurement to skin color for the condensation algorithm. The incorporation of Lucas Kanade tracking helps to increase the robustness of the condensation tracking algorithm.

In the third step, a Gaussian function is applied to smooth the weighted samples evaluated in the condensation algorithm. It generates a continuous 3D probability map, from which we could figure out the likelihood of the number and location of possible faces in every frame.

In the last step, a post-processing is conducted. As a result of our bi-directional tracking strategy, every frame has two possible sets of tracking results coming from the previous and the following seed frames respectively. The more reliable result is selected according to the cumulative Lucas-Kanade tracking error. Facial features of the same face are used to identify the corresponding face within a tracking sequence.

3.1 Face detection and skin color modeling

Recent research has discovered that human skin colors tend to be clustered in a transformed two-dimensional color space and the distribution of them can be represented by a 2D Gaussian function in the normalized color space [15]. In our system, we collect the skin color regions from various video frames and build a generic skin color model. We use the chromaticity transformation where the color pair (r,b) is defined as:

$$r = \frac{R}{R+G+B}, \quad b = \frac{B}{R+G+B} \tag{2}$$

We model the skin color distribution as $\mathcal{N}(m,C)$, where $x = (r\ b)^T$, $m = E\{x\}$ and $C = E\{(x-m)(x-m)^T\}$. Given a chromatic pair (r,b), the likelihood for it to be a skin color is defined as:

$$P(r,b) = \exp[-0.5(x-m)^T C^{-1}(x-m)] \tag{3}$$

In the first part of our face tracking system, we employ a well-established face detector developed in [1] to get the primal face location in every other I-frame. This face detector is able to detect frontal or slightly slanting faces with as least 32×32 pixel in size.

All the face regions detected are collected and the pixels in these regions are used to train a specific skin color model $\mathcal{N}(m',C')$. This model adapts to the lighting condition and the skin color of faces that are detected in this video sequence. This adaptive model makes it more accurate for feature tracking than the generic color model.

Since the face detector we are using may not be able to detect significantly tilted faces, or faces of very small size, there is a possibility that faces could not be detected in the I-frames. In this case, a generic skin color model will be adopted.

3.2 Tracking with multi-measurement condensation algorithm

In the second part of our system, we use the condensation algorithm to perform the tracking. Samples in the condensation algorithm are defined over 10X10-pixel windows in each frame.

The weight of a sample is the probability of it belonging to a face region.

In the prediction stage of condensation, the position of new samples are predicted by equation (1) in which the displacement c_t is estimated as:

$$c_t = C_{t-1} - C_{t-2} \quad (4)$$

where $C_t = \sum_{n=1}^{N} \pi_t^{(n)} s_t^{(n)}$, is the weighted average of samples in time-step t. The validity of this estimation is based on the assumption that the movement of faces in consecutive frames is minor and can be modeled as piece-wise linear.

In the measurement stage, we use the skin color model derived from the face regions, as well as the good features extracted from the detected face regions. We measure the skin color and facial features of all the samples.

As explained earlier, the pixels in all the detected face regions are used to train a specific skin color model $\mathcal{N}(m', C')$. For a single sample point $s_t^{(n)}$, which covers a 10 by 10 region, the skin color likelihood of this sample is:

$$l_c(s_t^{(n)}) = \frac{1}{100} \sum_{(r,b) \in s_t^{(n)}} P(r,b) \quad (5)$$

To complement the skin color measurement, we also measure the facial feature positions. For each of the detected faces in the seed frame, we divide the region into four equal sub-regions and select a good feature in each sub-region. We adopt the feature selection method introduced by [10].

These facial features are then tracked using the Lucas-Kanade algorithm [6]. At each tracking step, an error value is obtained for each feature to provide an indication of tracking confidence. If the error exceeds a given threshold, the tracking of this feature is taken as unsuccessful and the feature will be removed from the feature set. In measuring a sample $s_t^{(n)}$, we give high weightage to those samples with small tracking errors. We define the weight of $s_t^{(n)}$ as:

$$l_f(s_t^{(n)}) = \frac{1}{\min(e_t^i)|_{f_i \in R_t} + c_r} \quad (6)$$

where f_i is the i^{th} feature, e_t^i is the tracking error of f_i at time-step t, c_r is a constant introduced to avoid singularity.

After the two features are evaluated, we normalize the two likelihood values respectively as:

$$l_c'(s_t^{(n)}) = \frac{l_c(s_t^{(n)})}{\sum_{i=1}^{N} l_c(s_t^{(i)})}, \; l_f'(s_t^{(n)}) = \frac{l_f(s_t^{(n)})}{\sum_{i=1}^{N} l_f(s_t^{(i)})} \quad (7)$$

We then combine the two measurements as follows:

$$\pi_t^{(n)} = w_c l_c'(s_t^{(n)}) + (1 - w_c) l_f'(s_t^{(n)}) \quad (8)$$

Here we set w_c=0.5, i.e. set equal weight to both features. It could be easily verified that $\sum_{n=1}^{N} \pi_t^{(n)} = 1$. Until now, we have completed one iteration of the condensation algorithm. The output of this time-step t is $Output_t = \{ s_t^{(n)}, \pi_t^{(n)} \}$. The tracking procedure continues until the next seed frame or video boundary is met.

We stress that our system performs backward tracking as well as forward tracking from every seed frame. Thus, for every frame between two seed frames, there will be two tracking results coming from the previous and the next seed frame. The strategy to choose the more reliable result is presented in section 3.3.

3.3 Face localization

The output of each tracking step is a set of weighted samples. The weight of a sample indicates the probability of this sample falling within a face region. Given this information, we need to figure out where the faces are located.

As the data in $Output_t$ is discrete, we use a Gaussian kernel to construct a continuous probability map over the entire frame so that the positions of faces will be identified easily. The Gaussian function will distribute the weight of every sample to its neighborhood. The weight of any pixel in a frame is defined by:

$$w(u) = \sum_{n=1}^{N} \exp[-\frac{1}{2}(s_t^{(n)} - u)^T Cg^{-1} (s_t^{(n)} - u)] \quad (9)$$

where $u = (i, j)^T$ is the position in a frame with coordinate (i, j), and Cg is a covariance matrix. The value of Cg determines the range of pixels that are affected by a sample.

From the probability map $prob(x, y)$, we could identify the location and size of the faces. Figure 3 shows an example of a continuous probability map and its corresponding face locations:

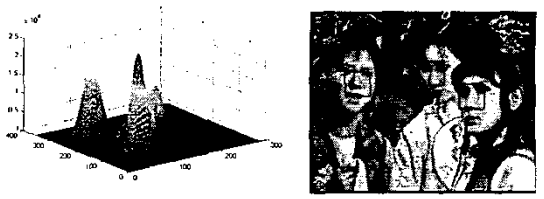

Figure 3: Probability map and corresponding face locations

3.4 Post-processing

Recall that our system has bi-directional tracking sequence. Therefore, there would be two sets of tracking results for every frame in-between two seed frames. We need to choose the more reliable result as the final output.

For any T-step tracking procedure starting from frame i (a seed frame), the tracking error of each step is calculated:

$$error_0 = 0; \; error_t = error_{t-1} + \sum_k e_t^k \quad (10)$$

where $t = 1, \ldots T$, and e_t^k is the tracking error of k^{th} feature in the Lucas-Kanade method.

For every frame, the tracking result associated with the smaller cumulative error is chosen as the final result. Each result may contain one or more face locations. These face locations are linked along the timeline into sequences of faces, denoted as $\{ S_1, S_2, \ldots S_Q \}$.

4. EXPERIMENT AND RESULT

We need to test the performance of our face tracking system on a variety of video clips. Since there is no standard evaluation suite, we selected 15 video clips from different sources to test our system. The video clips chosen are: 7 from movies; 2 from TV serial; 2 from MTV; 2 from documentary and 2 from live concert.

The average length of each test clip is 454 frames. The average face appearance is 1.22 faces per frame.

We adopt the following methodology to evaluate the accuracy of the face tracking results. Given that there are Q face sequences $\{S_1, S_2, \ldots S_Q\}$ tracked from a video clip, we define $L(S_i)$ as the length of face sequence S_i, i.e. the number of frames in S_i. If the location of the tracked face in L'_G frames of S_i overlap with the actual face, and $p_i = L'_G / L(S_i) > 60\%$, then we consider S_i to be a good tracking sequence.

Take L_T as the total length of the actual face sequences, and $L_G = \sum p_i L(S_i)$, where S_i is a good tracking sequence. Then the precision and recall of a tracking result can be calculated as follows:

$$\mathrm{Prec} = \frac{L_G}{\sum_{i=1}^{Q} L(S_i)}, \quad \mathrm{Recall} = \frac{L_G}{L_T} \tag{11}$$

we also define another figure of merit $F1$ as

$$F1 = \frac{2 \cdot \mathrm{Prec} \cdot \mathrm{Recall}}{\mathrm{Prec} + \mathrm{Recall}} \tag{12}$$

We compare our algorithm with two variants of our face tracking methods: one is to use only skin color as the measurement in condensation; the other is to track only facial feature points. The tracking results of the three methods are listed in Table 1.

Table 1: Comparison of the three algorithms

	Our method	Skin color only	Facial features only
Prec	87.33%	70.79%	86.26%
Recall	74.37%	59.58%	60.93%
F1	80.33%	64.70%	71.42%

From the table, we can see that our method could achieve an F_1 value of over 80%, which is significantly better than the other two methods.

The precision of the Lucas-Kanade method is lower than that of our method, owing to the false alarms from the face detector. Our method uses skin color and incorporates Lucas-Kanade tracking to feature point positions, so we could eliminate most of these false alarms.

The color-only method tends to find more face sequences than our method, but most of them are incorrectly located, especially when the background is cluttered. We can see from Table 1 that the recall of our method is higher than that of the color-only method.

5. CONCLUSION

This paper presented a robust face tracking system built on the condensation algorithm incorporating Lucas-Kanade feature tracking. Our method is suitable to video indexing and retrieval task. We could achieve an average F1 measure of over 80% on our test video set.

6. REFERENCES

[1] Chua T.S., Zhao Y. and Kankanhalli M.S. A compressed-domain face detection method for video stratification. To appear in Special Issue of *The Visual Computer*, Springer Verlag, 2002.

[2] Decarlo D., Metaxas D. The integration of optical flow and deformable models with applications to human face shape and motion estimation. In Proceedings of the Conference on *CVPR*, 231—238, 1996.

[3] Fieguth P., Terzopoulos D. Color based tracking of heads and other mobile objects at video frame rates. In Proceedings of the Conference on *CVPR*, Puerto Rico, 1997.

[4] Isard M., Blake A. CONDENSATION – conditional density propagation for visual tracking. *IJCV*, 1(29), 5-28, 1998.

[5] Leung T.K., Burl M.C., Perona P. Finding faces in cluttered scenes using random labeled graph matching. In Fifth *ICCV*, 637--644, 1995.

[6] Lucas B.D., Kanade T. An iterative image registration technique with an application to stereo vision. In Proceedings of the 7th International Joint Conference on *Artificial Intelligence*, 674—679, 1981.

[7] Maurer T., Malsburg C. Tracking and learning graphs and pose on image sequences of faces. In Proceedings of the *ICAFGR*, 176--181, 1996.

[8] Qian R.J., Sezan M.I., Matthews K.E. A robust real-time face tracking algorithm. In Proceedings of *ICIP*, 131 –135, 1998.

[9] Schwerdt K., Crowley J.L. Robust face tracking using color. In Proceedings of the 4th IEEE *ICAFGR*, 90 –95, 2000.

[10] Shi J., Tomasi C. Good Features to track. In Proceedings of the IEEE Conference on *CVPR*, 593--600, 1994.

[11] Spors S., Rabenstein R., Strobel N. Joint audio-video object tracking. *ICIP*, 393 –396, 2001.

[12] Strom J., Jebara T., Basu S., Pentland A. Real time tracking and modeling of faces: An EKF-based analysis by synthesis approach. Technical Report 506, M.I.T. Media Laboratory Perceptual Computing Section, 1999.

[13] Swain M.J., Ballard D.H. Color indexing. *IJCV*, 7(1), 11-32, 1991.

[14] Wang C., Brandstein M.S. A hybrid real-time face tracking system. In Proceedings of the IEEE *ICASSP*, vol. 6, 3737 – 3740, 1998.

[15] Yang J., Lu W., Waibel A. Skin-Color Modeling and Adaptation. In Proceedings of the 3rd *ACCV*, vol. 2, 687--694, 1998.

[16] Yuille A.L., Hallinan P.W., Cohen D.S. Feature extraction from faces using deformable templates. *IJCV*, 8(2) 99—111, 1992.