Face Tracking Using Motion-Guided Dynamic Template Matching

Liang Wang, Tieniu Tan, Weiming Hu
National Laboratory of Pattern Recognition
Institute of Automation, Chinese Academy of Sciences, Beijing, P. R. China, 100080
E-mails: {lwang, tnt, wmh}@nlpr.ia.ac.cn

Abstract

Combining two sophisticated techniques of motion detection and template matching, this paper proposes a simple but effective algorithm for detection and tracking of human faces. First, we use a statistical model of skin color and shape information to detect face in the first frame, and initialize it as an appearance-based intensity template for subsequent tracking. Second, incorporating background subtraction, projection histograms of moving silhouette and geometric constraints of body parts, we can quickly determine a good approximation of the search region corresponding to head location. Finally, a correlation-based template matching procedure is applied to further localize human face accurately, and current template can be dynamically updated in size and content to adapt temporal changes of the tracked face's scale and orientation. Moreover, a confidence measure representing the template's reliability is presented to guide possible template re-initialization for continuous face tracking. Experimental results demonstrate the validity of our proposed method.

1. Introduction

Image analysis of faces has been an active research topic in computer vision and image processing. This strong interest is driven by some promising applications such as surveillance and security monitoring, advanced human-machine interface, video conferencing and virtual reality. Generally speaking, major research areas include face detection, tracking and recognition, face animation, expression analysis, lip reading, etc. As the basis for all other related image analysis of human faces, face detection and tracking are of great importance.

Recently, there have been considerable research achievements in detection, recognition and tracking of human faces[1~12]. For example, Rowley et al.[1] proposed a neural network based algorithm for face detection, and Garcia and Tziritas[2] used quantified skin color regions merging and wavelet packet analysis in face detection. More face detection and recognition algorithms can be found in a review[3]. Instead of detecting human faces in each frame independently, face tracking utilizes temporal correlation to locate them. Presently, most researchers have emphasized on color-based[4,7~12] or model-based tracking[5~6]. For instance, Yang and Waibel[4] built a real time face tracking system based on the normalized color space, and Colmenarez et al.[5], DeCarlo and Metaxas[6] used a 3D face model in face tracking process. However, these algorithms seldom deal with multiple faces effectively, especially occlusion. Furthermore, these algorithms are usually computationally complex due to their use of color correlation, blob growing, Kalman filter prediction, 3D model, etc.

In this paper, based on an efficient combination of motion detection and template matching, we develop a much simpler algorithm for face tracking which achieves highly satisfactory tracking performance. It is well known that the tracking properties of motion detection and template matching are complementary. In other words, moving targets can be precisely tracked using dynamic template matching guided by motion detection. Therefore, this motivates us to present a simpler procedure for face detection and tracking. It can be simply described as follows: first, a statistical model of skin color and shape information are used to extract the face in the first frame, and the detected face is initialized as an appearance-based intensity template for later tracking; second, incorporating background subtraction, projection histograms of moving silhouette and structural constraints of body parts, we can determine a suitable search region corresponding to an approximation of head location quickly; finally, a correlation-based template matching procedure is applied to find the best match as the refined face position, and the current template can be dynamically updated in size and content to cope with temporal changes of the face's scale and orientation. Experimental results demonstrate the algorithm's effectiveness in face detection and tracking.

The main contributions of our algorithm include: 1) The use of motion detection in guiding correlation-based template matching prevents the template drifting onto background; 2) The use of computationally expensive Kalman filtering, 3D model, or other probabilistic approaches is avoided; 3) Incorporating projection histograms of moving silhouette with constraints on body parts can significantly narrow down the search range; 4) Except for template initialization, the proposed algorithm is mostly performed on the transformed grayscale images, which increases the likelihood of real-time face tracking.

The remainder of this paper is organized as follows. Section 2 outlines the algorithm of face tracking. Face detection using the skin color model and shape information is introduced in Section 3. Section 4 describes the face tracking process, including motion segmentation, determination of search regions, and template matching and updating. Experimental results are presented and discussed in Section 5.

2. Overview of our algorithm

Our face tracking system shown in Figure 1 consists of three major processing modules: template initialization, determination of search regions, and dynamic template matching and updating. Template initialization can be thought equivalently as a face detection problem. In our approach, we choose the *normalized color space*[4] to represent skin-like regions. The parameters of the skin color model can be estimated using the *maximum likelihood estimation* (*MLE*) method. Once human face is detected, it will be converted into an appearance-based intensity template for subsequent tracking. Using background subtraction, regions of change can be fast segmented from the background constructed by *the least median of squares (LMedS)* method[13]. Meanwhile, *projection histograms* of moving silhouette and *body parts constraints* are utilized to determine the corresponding search space. In the last module, a template matching process based on *the sum of squared difference* (*SSD*) distance measure between the template and a search region is performed to accurately localize face position in current image. Also, the template is dynamically updated using an *infinite impulse response* (*IIR*) filter[16] to adapt to its size and content changes in response to face movement along sequential frames. In addition, we define a *template confidence measure* to bootstrap the re-initialization of the template for continuous and reliable tracking.

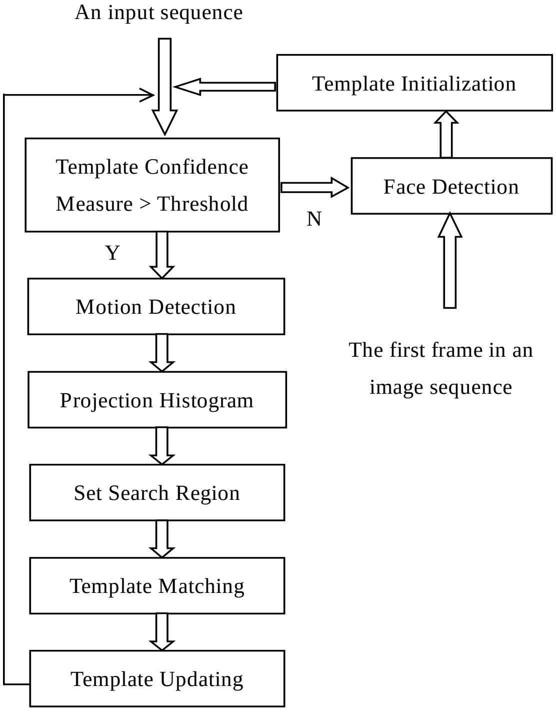

Figure 1. Block diagram of face tracking algorithm

3. Face detection

As far as face detection is concerned, many approaches have been proposed using texture, shape, and color information or their combinations[2,4,7~12]. For simplicity, we only use color and shape information to realize face detection.

3.1. Skin Color Model

Color is a simple but important pixel-based feature to detect human faces. At present, there have been various color representations for skin. For example, Ohya et al.[9] represented skin region using *LUV* color space, and Sobottka and Pitas[11] used *HSV* color space and shape information to extract facial regions. In this paper, we choose the simple and efficient *normalized r-g color space* described by Yang and Waibel[4]. In their studies of skin-color distributions, three conclusions could be obtained[4]: a) *skin color distributions of different people are clustered in a relatively small area within the chromatic space*. That is, skin colors of different people are very close, but they differ distinctly in intensities; b) *skin color differences among different people can be reduced by intensity normalization*; c) *under certain lighting condition, skin-color distribution can be characterized by a multivariate normal distribution*. Assuming that a face color distribution is represented by 2D *Gaussian model* N (M, $\sum^2$), we can learn the parameters of the mean and covariance matrix using *Maximum Likelihood Estimation* (*MLE*).

$$M = (r_m, g_m)^T = \frac{1}{N}\sum_{i=1}^{N} X_i \tag{1}$$

$$\Sigma = \frac{1}{N}\sum_{i=1}^{N}(X_i - M)(X_i - M)^T \tag{2}$$

Where X_i, N and the superscript T are the ith sample, the total number of training samples and transpose, respectively. The procedure for creating skin color model can be described as follows[4]: 1) take a set of images and select skin-colored region interactively; 2) estimate the mean and the covariance of the color distribution in the normalized color space; 3) substitute the estimated parameters into the Gaussian distribution model. Since the model only has six parameters, it is easy to adjust them to different people and lighting conditions.

3.2. Face template initialization

First, the normalized *r-g* values of each pixel *i* in an input image are converted into the color distance from standard skin color, which is measured by *Mahalanobis distance* denoted by the following equation, where the vector $X_i=(r_i, g_i)^T$.

$$D_{(i)} = (X_i - M)^T \Sigma^{-1} (X_i - M) \qquad (3)$$

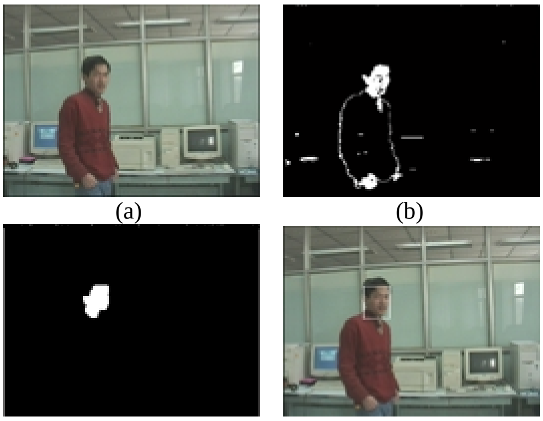
(a) (b)
(c) (d)

Figure 2. Face detection

(a) An original image; (b) The result after skin-color filtering; (c) The result after further shape filtering; (d) The final result.

Then, we make a histogram of color distances from previous results to extract skin-like regions by a discriminant analysis. Finally, some additional operations such as size and shape based filtering are performed to further extract face regions with appropriate size. As we know, it is not always easy to locate human faces because the background may contain other skin-like colors too. So, we should perform some post-processing to eliminate some isolated skin-like pixels and small blobs such as hands. First, we use connectivity and morphological filtering to eliminate some isolated points and noises. Then, we apply a connected component analysis at a coarse resolution. Finally the geometric size and shape features of human face are analyzed to further remove some non-face small blobs. The decision criteria rely on combinations of the area *A*, the perimeter *P*, and the size (D_x, D_y) of the bounding box of the connected component[10]. Several measures of shape information are defined as follows, and all regions with any of *C*, *S* and *O* lower than the corresponding predefined thresholds are removed.

$$C = \frac{A}{P^2}, \quad S = \frac{A}{D_x D_y}, \qquad O = \frac{D_y}{D_x} \qquad (4)$$

An example of face detection using color and shape information is shown in Figure 2, from which we show that the tie similar to skin-color and both hands are effectively removed. The detected face is initialized as an *intensity template* for subsequent tracking.

4. Face tracking

We utilize the *coarse-to-fine* strategy for face tracking. First, we determine an approximation of search region at a large scale corresponding to head position through motion detection. Then, the dynamic template representing temporal changing information among image frames is matched in search regions to further localize face position accurately.

4.1. Determination of search region

The determination of search regions aims at speeding up later tracking process. It involves the following three steps.

a) *Motion Detection*: Background subtraction is a particularly efficient method for detecting gray level changes. Generally, it is composed of the generation of the background, the arithmetic subtraction operation and the selection of a suitable threshold. A potentially robust approach is to dynamically generate background from some portion of image sequence and periodically update it to account for possible changes in the background. The *least median of squares* (*LMedS*) method is adopted[13]. Let I_{xy}^t represent a sequence of *N* collected images, and (*x*, *y*) is the pixel location. The resulting background B_{xy} can be computed by

$$B_{xy} = \min_b med_t (I_{xy}^t - b)^2 \qquad (5)$$

where *b* is background value to be determined. Then, a threshold image T_{xy} is determined from the *median absolute deviation* at each pixel: T_{xy}=2.5×1.4826×MAD_{xy}, where MAD_{xy}=med_t $|I^t_{xy}$-$B_{xy}|$ and the constant 1.4826 is a normalization factor with respect to a Gaussian distribution[13]. Finally, the difference image D_{xy} can be obtained by comparing difference values between the incoming image and the background against the threshold image.

$$D_{xy} = \begin{cases} 1 & | I_{xy}^t - B_{xy} | \geq T_{xy} \\ 0 & Otherwise \end{cases} \qquad (6)$$

The segmented foreground regions probably lead to spurious pixels, holes inside moving objects and other anomalies, therefore they need to be further filtered using *morphological* operators. Finally, a *binary connected component analysis* is applied to clearly extract highly concentrated moving regions. An example of change detection is shown in Figure 3 (a)~(d).

b) *Projection Histogram*: Projection histogram has been shown to be useful in some computer vision systems. For example, Kuno[14] used shape features of moving silhouette patterns as the parameters to detect human in surveillance system, and these shape features are mainly the mean and the standard deviation of projection histograms of silhouette patterns. Also, W^4 system[15] determined whether the foreground region contained multiple people by analyzing vertical projection histogram of silhouettes. We also utilize projection histogram to represent the shape of a binary silhouette. The horizontal and vertical projection histograms, which will be used to determine search regions combining constraints on body parts, can be easily computed by projecting the foreground region on an axis perpendicular to major axis and along major axis of human body, respectively. Figure 3 (e)~(g) give an example of projection histograms of moving silhouette.

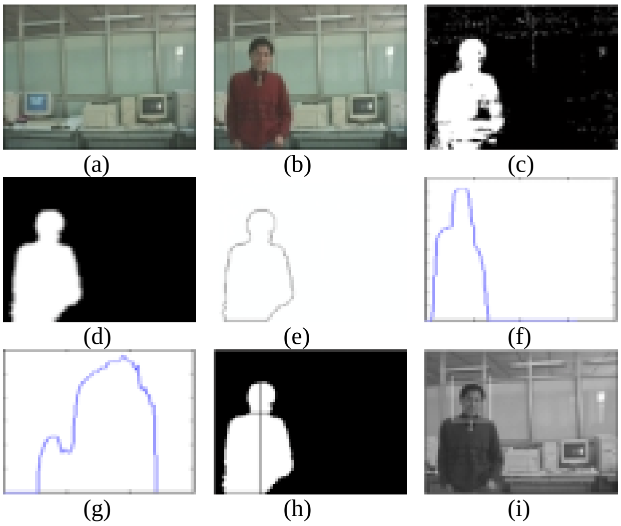

Figure 3. Determination of search region

(a) The background image constructed by *LMedS*; (b) The original image; (c) The result after background subtraction; (d) The result after post-processing; (e) Moving silhouette; (f) and (g) The vertical and horizontal projection histograms of moving silhouette; (h) The approximate positions of major axis and the neck of human body; (i) The search region superimposed by a white rectangular box.

c) *Adding Body Constraints*: The shape of human body is symmetrical about its major axis, and the head usually appears above the trunk. Therefore, this topological structure will show some specific properties in projection histograms of its silhouette. We guess that the positions of body major axis and the neck should correspond to *a local peak* in the vertical projection histogram and a *local valley* in the horizontal projection histogram, respectively. The major axis is determined by finding a local peak higher than a threshold value selected as the mean value of the entire histogram in the vertical projection histogram[15]. Similarly, we can determine the neck coordinate by searching a local valley in the horizontal projection histogram. As we know, the aspect ratio of human face is about 1/1.2. Based on these assumptions, we can initially set a search region as an approximate head location. Allowing for motion segmentation errors, search regions should be suitably adjusted. On the one hand, if it is set too small, it may not completely include the tracked face. On the other hand, if it is set too big, it will accordingly increase the computational cost during the matching process. For accurate tracking, we expand the search range 1.5 times the initially determined head region using projection histograms. A result of the determination of search region is shown in Figure 3 (h)~(i).

4.2. Template matching and updating

Generally, correlation matching is computationally the most expensive part of the tracking algorithm. However, if it only need be performed in a relatively small search region, the computation time will decrease significantly. Also, template matching is biased towards areas where motion is detected, so it is more likely to prevent the template drifting onto background.

a) *Correlation-based Matching*: All candidate search regions are used to match with the current template so as to find the best matching result, and several special details should be considered: 1) because of periodical background updating, a human will be probably missed out by motion detection when he or she stops for a while. So, an extra search region should be used, which is composed of the pixels in current frame in response to the location of $CR_{n\text{-}1}$; 2) it is reasonable to assume that the location and size of faces within each frame do not change much. Therefore, to improve tracking speed, we only need to search possible faces in neighboring search regions guided by the displacement information between search regions' centers; 3) in general, there will be a possible transient overlapping between two walking figures. If their heads are isolated clearly, we can still accurately determine their corresponding search regions by projection histograms. If their heads are partially occluded, motion detection can only provide a composite search region that will be used to match all templates to find one most suitable match. Certainly another occluded face can still be tracked approximately using motion speed information of the face and its template is updated until the transient occlusion ends.

To maintain a lock on the tracked object, the underlying correlation algorithms must be robust to scale changes. This can be solved by selective template size adjustment before the matching process. For each position of search region, a similarity measure between the template and the search block is evaluated by the *sum of the squared difference* (*SSD*) of grayscale value. For a displacement (u, v) between reference template and search block, the difference at pixel position (x, y) can be represented by

$$D_{xy}(u,v)=\sum_{x,y}(S(x,y,t)-T(x+u,y+v,t-1))^2 \qquad (7)$$

where S (x, y, t) denotes the grayscale at position (x, y) at time instance t, and T represents the template. The best match position is found by choosing the minimum difference value greater than a predefined threshold.

b) *Template Updating*: The correlation process has to deal with incremental changes of the tracked object's appearance, so it is expected that the template be updated to catch up with the smooth variation of face appearance from one frame to another. In this paper, we use an *infinite impulse response* (*IIR*) filter to update the template[16]. Once a best correlation match at the search block M_n in current frame is found, it will be merged with previous template T_{n-1} through an IIR filter to produce a new template T_n for tracking in subsequent frames. This process of template updating can be formulized by

$$T_n = \alpha M_n + (1-\alpha)T_{n-1} \qquad (8)$$

where α is a time constant that specifies how fast new information replaces old observations. An example of template updating using *IIR* filter is shown in Figure 4.

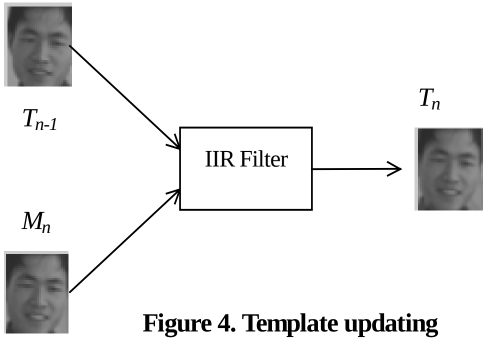

Figure 4. Template updating

c) *Template Confidence Measure*: Many factors such as the accumulated errors of segmentation and frequent adjustments of template size maybe affect tracking process. To track human faces continuously and reliably, we define a *confidence measure* to observe the reliability of the template. It is proportional to the resolution of the template and the matching value of similarity measure function in template matching process. Once it is lower than a suitable predefined threshold, that is, it is at too low resolution to represent real image content of the tracked face, skin-color based template initialization must be bootstrapped again. Furthermore, if a new human is detected by motion segmentation, a new template initialization should be performed accordingly for his or her face tracking.

5. Experimental results

Many experiments have been carried out to evaluate the tracking performance. In our laboratory with a relatively complex background, the observed people are required to move randomly in the viewable area, and a digital camera Panasonic NV-DX100EN fixed on a tripod is used to capture a group of true-color image sequences with the resolution of 640×480. These images are then sub-sampled into the resolution of 320×240. Our proposed algorithm is mainly performed in two cases. One is single face tracking, another is multiple faces tracking even in the presence of possible occlusion.

For single face tracking, our algorithm provides a robust tracking result even in the presence of slight scale and orientation changes of human face. Figure 5 shows an example of single face tracking, where the bigger and smaller white rectangular boxes represent the corresponding search region and the tracked face respectively. From Figure 5, we can show that the tracking results are encouraging though the continuous smooth changes of the face's size and orientation exist. This benefits greatly from the precise determination of search region and dynamic updating of template in response to face movement frame by frame.

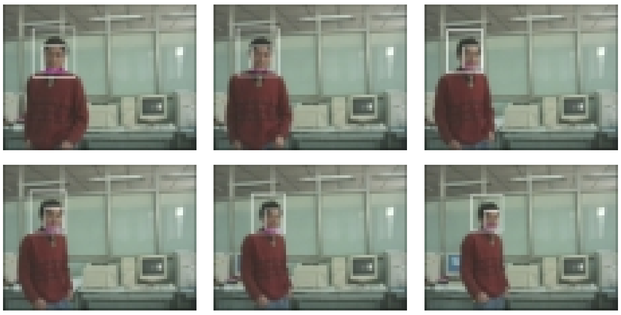

Figure 5. Single face tracking

Tracking of multiple faces is clearly more difficult because these faces may occlude each other. Without loss of generality, we consider the tracking of two faces in our experiment. When the faces move in their own trajectories independently, we can track them separately with multiple dynamic templates. When two human bodies meet but their heads are separated, the determination of search regions is still successful. When head occlusion happens, the difficulties of multiple face tracking will depend on several factors, such as how similar these faces are, how long the occlusion lasts, and at what percentage one face is occluded by another face. We assume that occlusion is partial and transient. When the occlusion happens, motion segmentation only provides a composite search region. Therefore, the non-occluded face can still be tracked by template matching, while the tracking of the occluded face will possibly be a temporal failure. However, we may turn to motion information to approximately estimate its position. Generally, it is reasonable to assume that the location and size of faces within each frame do not change much. Therefore, we may simply utilize motion information of face speed with respect to the changing distance between search regions' centers among the neighboring frames to approximately predict the occluded face's position in next frame, and its template is updated until the transient occlusion ends. Figure 6 gives a successful example of two faces tracking in the presence of occlusion, where only the tracking result of the person in jacket is shown for clarity. From Figure 6, we can see that the tracking results before, during and after occlusion are very precise.

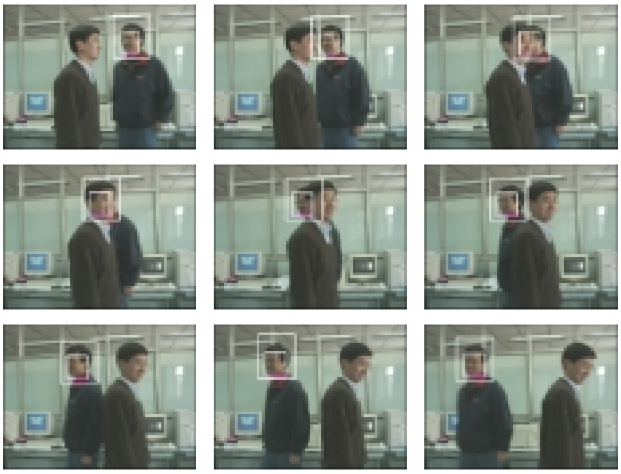

Figure 6. Face tracking in the presence of occlusion

Of course, our algorithm will probably fail under some special conditions such as the sudden significant changes of size and orientation of face movement. This is because that if face movement is too sudden, the template matching and updating will be unreliable for representing face-changing information. In a word, a large number of experimental results demonstrate that our tracking algorithm obtains highly satisfactory performance.

6. Conclusions

In this paper, a novel face-tracking algorithm based on an efficient combination of template matching and motion detection is proposed. Using motion detection to guide template matching can not only prevent the template drifting into background, but also provide fast and robust tracking despite occlusion without the requirement of having a temporal prediction filter such as Kalman filter. We test the algorithm on some video sequences, and experimental results show our algorithm works well. There are a number of directions to improve the algorithm. For example, we can choose better motion detection methods to decrease segmentation error, and use more motion information to effectively cope with occlusion in multi-face tracking in future work.

Acknowledgements

The authors would like to thank Ms. Y. C. Fang for her valuable discussion. This work is supported in part by NSFC (Grant No. 58925105), and the Innovation Fund of Institute of Automation (Grant No. 1M01J02), Chinese Academy of Sciences.

References

[1] H. Rowley, S. Baluja, and T. Kanade, "Neural Network based Face Detection", *PAMI*, 1998, pp. 23-38.

[2] C. Garcia and G. Tziritas, "Face Detection Using Quantified Skin Color Regions Merging and Wavelet Packet Analysis", *IEEE Trans. on Multimedi*a, 1(3), 1999, pp. 264–277.

[3] A. Samal and P.A. Iyengar, "Automatic Recognition and Analysis of Human Faces and Facial Expressions: A Survey", *Pattern Recogniti*on, 25(1), 1992, pp. 65–77.

[4] J. Yang and A. Waibel, "A Real-Time Face Tracker," *WACV*, 1996, pp. 142-147.

[5] A. Colmenarez, R. Lopez, and T. Huang, "3D Model-Based Head Tracking", *VCIP*, CA, 1997.

[6] D. DeCarlo and D. Metaxas, "Deformable Model-Based Face Shape and Motion Estimation", *ICFG*, 1996.

[7] A. Saber and A.M. Tekalp, "Frontal-view Face detection and Facial Feature Extraction Using Color, Shape and Symmetry Based Cost Functions", *Pattern Recognition Letter*s, 19(8), June 1998, pp. 669–680.

[8] Gang Xu and T. Sugimoto, "A Software-based System for Real-time Face Detection and Tracking Using Pan-Tilt-Zoom Controllable Camera", *ICPR*, 1998, pp. 1194-1197.

[9] M. Ohya et al, "Face Detection System by Using Color and Motion Information", *ACCV*, 2000, pp. 717-722.

[10] B. Menser and M. Wien, "Segmentation and Tracking of Facial Regions in Color Image Sequences", *VCIP*, vol. 4067, Perth, Australia, June 2000, pp. 731–740.

[11] K. Sobottka and I. Pitas, "A Novel Method for Automatic Face Segmentation, Facial Feature Extraction and Tracking", *Signal Processing: Image Communicatio*n, 12(3), June 1998, pp. 263–281.

[12] P. Fieguth and D. Terzopoulos, "Color-based Tracking of Heads and Other Mobile Objects at Video Frame Rates", *CVPR*, 1997, pp. 21-27.

[13] Y.H. Yang and M.D. Levine, "The Background Primal Sketch: An Approach for Tracking Moving Objects", *Machine Vision and Applications*, 5, 1992, pp. 17-34.

[14] Y. Kuno et al, "Automated Detection of Human for Visual Surveillance System", *ICPR*, 1996, pp. 865-869.

[15] I. Haritaoglu, D. Harwood, and L. Davis, "W4: Real-Time Surveillance of People and Their Activities", *PAMI*, 22(8), 2000, pp. 809-830.

[16] A Lipton, H. Fujiyoshi, and R. Patil, "Moving Target Detection and Classification from Real-time Video", *WACV*, 1998.