On Face Tracking in Video Sequences

Cornélia Janayna Pereira Passarinho, Evandro Ottoni Teatini Salles and Mario Sarcinelli-Filho
Departamento de Engenharia Elétrica, Universidade Federal do Espírito Santo
Vitória, ES, BRAZIL
{janayna; evandro; mario.sarcinelli}@ele.ufes.br

Abstract—This paper proposes a framework to track faces in image sequences in a video. The technique combines face detection through using Support Vector Machine (SVM) with target tracking through using a Kalman filter. The adjacent locations of the target point are predicted in a search window that reduces the number of image regions that are candidates for faces. Thus, the method can predict the object motion more accurately. Our architecture is distinguished by the satisfactory results for videos containing scale variation, bad illumination and complex background. Brightness compensation is applied to improve the detection of faces in videos.

Keywords-face detection; face tracking; adaptive face tracking;SVM.

I. INTRODUCTION

Human face detection and tracking plays an important role in many applications, such as video surveillance, face recognition, and face identification [1]. Several researchers have detected face by combining color-based methods to obtain high performance and high speed [2]. The advantages are that such methods are fast and have high detection ratio. However, these typical methods have some drawbacks, since color-based methods are limited in the presence of varying lighting. Furthermore, the detector can find objects having a color that is similar to the one of the target. Other researchers have used eye features, such as eyeball, the white part of the eye, or even the pupil. However, it results in false detection when the person closes the eyes or wears glasses.

Many papers present feature-based methods to detect faces [3] [4]. Specifically, feature-based detection demands huge computational effort and low-speed operation. Moreover, the main problem of this approach is that it requires an eye detector, a nose detector, a mouth detector and so on. In those cases the problem of detecting faces has been replaced by the problem of detecting multiple, similarly complex and deformable, parts. Although the set of detectors can reduce the accuracy required for each facial feature detector, the system should perform multiple detection tasks [Pisarou]. Such methods are useful for facial analysis and correspondence in face identification, because detection and alignment of facial features demands images of relatively high spatial resolution. However, in dynamic scenes, face detection often needs to be achieved at much lower resolution. Occlusions caused by changes in the viewpoint are the main problem with the local feature-based approaches. In such case, correspondences between certain features do not exist under occlusion.

In this paper, we propose a face detection and tracking algorithm that can detect human faces under poor lighting conditions and different views. We do not use face color model or deformable face parts to find faces in a video. Instead, face image is the feature considered for SVM training. The location of the face is estimated in a search window in each frame, by using a Kalman filter. The prediction function of the Kalman filter decreases the area to be searched, thus increasing the tracking rate and also enhancing the tracking performance. We also use lighting compensation to improve the performance of the framework. The result is a method that is effective under facial changes, such as eye-closing, glass-wearing or emotions, under faces having distinct profiles and under bright variation. It is also worth to emphasize that the tests here presented were performed on poor resolution video sequences.

The paper is hereinafter split in some sections, to address the above mentioned topics. Section II describes the approach used in preprocessing scenes, and Section III presents Gabor Features. In the sequel, Section IV presents the use of SVM to detect faces. Kalman Filter is briefly described in Section V, and then the complete adaptive face tracker is shown in Section VI. The relevant results and conclusions are presented in Sections VII and VIII, respectively.

II. DYNAMIC SCENES

A. Face Detection

The performance of face-tracking in dynamic scenes is quite dependent on the accuracy of the target-detection step. First, a method to circumvent the presence of poor lighting should be applied. Besides, the classifier used for face detection should achieve low false-positive rates. This is extremely difficult because there are image regions that, when observed out of their whole context, appear as face-like regions. In spite of this, it is worth to note that the number of false-positives can be reduced by searching only the image regions where it is likely to find face pixels.

Skin color filter is considered as an important method for removing non-face pixels [2]. Skin color filters can rapidly remove non-skin color pixels, which reduce the search region for the face detection step.

In this work we provide a pre-processing method to achieve a high performance in face detection. First, the light compensation algorithm presented in [5] is applied. Next, we use a detection algorithm based on skin-color regions to improve the performance of the tracking step.

B. Image Preprocessing

We have implemented a two stages method in which RGB is compensated as follows. The algorithm presented in [5] for this propose is based on

$$S = \frac{C_{std}}{C_{avg}} \tag{1}$$

where S is a scale factor for one specific color channel (R, G or B). C_{std} and C_{avg} separately are the standard mean gray value and the mean value of the specific channel. C_{avg} is the mean value of the non-black pixels in each channel.

At the second stage, the algorithm in [6], proposed to detect the skin region in a color image, uses the constraints on the RGB values of each pixel in the image to identify the skin regions. The thresholds applied to each image pixel are 95 for R, 40 for G and 20 for B. Next, if the absolute difference between the R and G values is higher than 15 and the R values are higher than both G and B band values, the pixel is then classified as skin. However, it is worth mentioning that this method did not perform well without the previous stage of light compensation.

III. GABOR FILTER

In this paper, face and non-face preprocessed images are presented to a classifier (a Support Vector Machine). Actually, such classifier receives global features obtained using the Gabor filter (Gabor features are effective to 2D object detection and recognition, as shown in [7]).

Gabor filters are defined by

$$\psi(\mathbf{x}) = \frac{1}{2\pi\sigma^2}\exp\left[\left(\frac{-\|\mathbf{x}\|^2}{2\sigma^2}\right)\right]\exp\left[j2\pi(\mathbf{w}^T\mathbf{x}+\phi)\right] \tag{2}$$

where $\mathbf{x} = (x, y)^T$, $\phi = \mu\pi/4$, $\mathbf{w} = (U, V)^T$, $\mu = 0,\cdots,3$, and $\sigma = \pi$.

There are some results showing that Gabor features of only one frequency level lead to a good performance in face recognition [7]. Therefore, in the following experiments, Gabor filters of four different orientations with one frequency level are used to speed up the recognition task. The sizes of the Gabor filters were set to 30 × 40 pixels. The positions which give large Gabor outputs are different depending on the orientation parameter ϕ of the Gabor filter. Thus, Gabor properties are suitable to enhance the different target poses in a video sequence.

IV. SUPPORT VECTOR MACHINE

SVM determines the optimal hyper plane which maximizes the distance between the hyper plane and the nearest sample, called margin [7]. When the training set (sample and its label) is denoted as $S = ((x_i, y_i),\cdots,(x_L, y_L))$, the optimal hyper-plane is defined by

$$f(x) = \sum_{i\in SV}\alpha_i y_i K(x_i, x) + b \tag{3}$$

where SV is a set of support vectors, b is the threshold and α is the solution of a quadratic programming problem. The training samples with non-zero α are called support vectors. $K(x_i, x)$ is the inner product $\Phi(x_i)^T\Phi(x)$ between the support vector x_i and the input vector x in high dimensional space. In our implementation, normalized polynomial kernel is adopted as the kernel function, which is defined as

$$K(x, y) = \frac{(1 + x^T y)^d}{\sqrt{(1 + x^T x)^d (1 + y^T y)^d}}. \tag{4}$$

In this paper we chose the well known and tested LibSVM library [8] to perform face detection, adopting the following parameters: normalized polynomial kernel and cost misclassification parameter C = 0.01, chosen after a set of experiments.

V. KALMAN FILTER

Kalman filter [9] is the adaptive linear filter most commonly used to solve problems of optimum estimate. This filter is the optimal solution to the tracking problem presented here. By using the Kalman filter the posterior location of the face in the frame is predicted based on the current position information. This step avoids the face search to be accomplished in the entire image.

At each time instant k it is supposed that the face is moving with constant velocity. The face motion model used in our tracking method is defined by the follow set of space-state equations:

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{v}_k \tag{5}$$

$$\mathbf{z}_{k+1} = \mathbf{H}\mathbf{x}_k + \mathbf{w}_k \tag{6}$$

where x_k represents the state vector characterized by its position and velocity, with transition matrix $\mathbf{A} = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$, and $z_k \in R^2$ represents the face position observed with observation

matrix $\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$. The transition ($\mathbf{v}$) and measurement ($\mathbf{w}$) noises are assumed white noises.

VI. Adaptive Tracking

In this paper, the detector and the tracker are used simultaneously. In this section we address the way they cooperate with each other. The complete face tracker algorithm is as follows:

1. In the first frame, as a previous estimate of the face position is not available, the face is searched for in all the image regions of skin color. The face thus detected becomes the current observation for the Kalman filter, and is obtained by using the SVM in each skin color region. Therefore, the skin color surrounding the face is all the regions of skin in the frame;

2. The estimate of the face location for the next frame is then estimated by using the Kalman filter;

3. A new observation is obtained in the skin color vicinity centered in the position estimated in the previous step, using SVM again;

4. If the target is detected in the region of interest, the algorithm returns to step 2. However, if the target is not detected in such region, the algorithm returns to step 1, trying to get a new initial observation.

VII. Results

In this section the effectiveness of the proposed adaptive tracking method is checked. The size of the images used is 240×320 pixels. Test video sequences are captured using a webcam. The total number of frames is 5720, in six video sequences. For training the classifier, we use the face and non-face images taken from videos and some face databases used in [2]. The face regions of these images are cropped by using the positions of nose. Examples of face images are shown in Figure 1, where the image size is 30 × 40 pixels. In the sequel, four Gabor features are obtained from each image. Next, we prepare the face and non-face images for training the SVM. In this experiment, 30×40×4 dimensional Gabor features are used, and SVM is applied to each one. Exhaustive search only in the skin color regions is used in the first frame, since the position of the target face is unknown. From frame 2 on, the proposed tracking method is applied to the vicinity of the target position in the previous frame. In our experiments, the vicinity region is just the set of skin color pixels. Red rectangles show the tracked region, giving the estimate for the face location, which is given by the correction step in the Kalman filter.

In this work, we show the results of the face detection and tracking corresponding to three of the captured video sequences. The video sequences were chosen for presenting more complex movement, scale variation, partial occlusion, appearance changing (glasses wearing and not wearing) and face view changing.

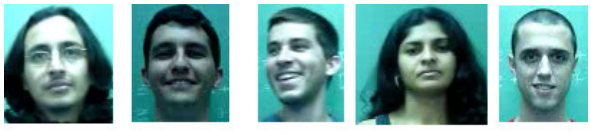

Figure1. Example of face images.

In the video sequence 1 a man walks to the right and to the left, always facing the camera. He speaks, stops, laughs, puts his eyeglasses and removes them. He also walks towards the camera, thus causing scale changes. Figure 2 shows some snapshots with the face tracking results for this video sequence. In this sequence face detection and tracking are submitted to image modifications due to scale and light changes and also to the appearance (eyeglasses). The robustness of the tracking method is noted by observing the frames 1, 220, 300 and 2000 of the video. After some time the man, in the frame 220, walks towards the camera, and returns to the original position some frames after (in the frame 300). The final test in this sequence is the appearance change. In most frames the man had the eyeglasses on the face. However, even after the eyeglasses were removed (frame 2000), our method was able to correctly detect and track the target face.

We also checked the effectiveness of the proposed method under face rotation and partial occlusion. The video sequence used in the test was the number two, for which six snapshots are shown in Figure 3, with the face detected shown as a red rectangle. From the frame 2 to frame 60 the man moves in front of the camera. He modifies his position from one side of the image frame to the other. After some frames he also moves his body in the vertical direction (frame 200 to frame 300). Observe also that he modifies his face view: in frame 200 the man outlines a partial profile, and then he moves his face until getting a new frontal view in frame 300. Starting in frame 1000 the man moves the face until reaching a full profile, in frame 1300. As shown in the snapshots of Figure 3, even with all these face rotations the man's face is correctly detected and tracked in this video.

The last test was performed using the third video sequence (four snapshots are shown in Figure 4, where

Figure 2. Four snapshots (frames 1, 220, 300 and 2000 in a zig-zag sequence) of the face tracking for the video sequence 1, showing the scale and light changes.

Figure 3. Six snapshots (frames 2, 60, 200, 300, 1000 and 1300, in a zig-zag sequence) of the face tracking in the video sequence 2, which shows face rotation and vertical displacement.

the red rectangles mark the detected faces once more). In this sequence we check the performance of the proposed system under partial face occlusion. Our method correctly detects the face from frame 69 (frontal face view) until frame 200 (inclined face), as shown in the two first snapshots in Figure 4. Unfortunately, the method fails in the frame 261 where the person has inclined her face and the light has changed (third snapshot in Figure 4). However, just a frame after, in the frame 262, the method correctly detects the target face again (frame not shown in Figure 4). There, as in the vicinity of the estimated face location no face was detected, the target should be searched for in the complete skin color region, thus correcting the error in the previous frame. Then, the target face is correctly tracked for the following frames, even under partial occlusion (frame 298, the fourth snapshot in Figure 4).

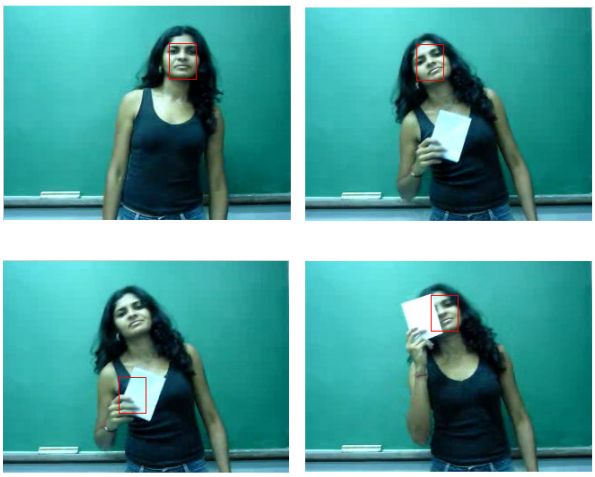

Figure 4. Four snapshots (frames 69, 200, 261 and 298, in a zig-zag sequence) for the face tracking in the video sequence 3, showing partial face occlusion.

Finally, it should be stressed that in spite of all image changes present in the video sequences used, due to body movements, light intensity changes, and even partial occlusion, the tracking method here proposed was able to effectively track the face of a person.

VIII. Conclusions

In this paper we propose an efficient face detection and tracking method. Such approach has shown good results in poor resolution videos captured with a webcam. The fluorescent lighting used in the scene resulted in poor perception for some colors. However, the preprocessing step we used is able to compensate for those effects. As a consequence, the faces were correctly detected in the test videos. The main contribution of the work is the improvement of face tracking by associating face detection along with next face position estimation based on Kalman filtering. As future work, we intend to use the method in outdoor videos.

Acknowledgment

The authors would like to thank CAPES, for the financial support, and the Graduate Program on Electrical Engineering of UFES, for providing the necessary support for the development of this research.

References

[1] S. Gong, S. McKenna, A. Psarrou, Dynamic Vision from Images to Face Recognition, 1st ed., Imperial College Press: Clarendon, 2000.

[2] M. H. Yang, D. J. Kriegman and N. Ahuja, “Detecting faces in images: a survey,” IEEE Transaction on Pattern Analysis and Machine Intelligence. vol. 24, pp. 34–58, January 2002.

[3] B. Castañeda Y. Luzanov and J. C. Cockburn “Implementation of a modular real-time feature-based architecture applied to visual face tracking.,” in Proceedings of the 17th International Conference on Pattern Recognition, pp. 167–170, Portugal, 2004.

[4] J. Ruan and J. Yin, “Face detection based on facial features and linear support vector machines,” in Proceedings of the International Conference on Communication Software and Networks, pp 371-375, 2009.

[5] Y. T. Pai, S. J. Ruan, M. C. Shie, and Y.C. Liu, "A Simple and accurate color face detection algorithm in complex background," IEEE International Conference on Multimedia and Expo, pp.1545-1548, July. 2006.

[6] Gayathri. Face: A skin color Matlab code. Software available at http://www.mathworks.com/matlabcentral/fileexchange/24851-illumumination-compensation-in-rgb-space?controller=file_infos&download=true, 2001.

[7] K. Hotta, “Adaptive weighting of local classifiers by particle filters for robust tracking,” Patter Recognition, vol. 42, pp. 619-628, May 2009.

[8] C. C. Chang and C. J. Lin. LIBSVM: A library for support vector machines.Software available at http://www.csie.ntu.edu.tw/˜cjlin/libsvm, 2001.

[9] O. Boumbarov, S. Sokolov, P. Petrov, A. Sachenko and Y. Kurylyak, “Kernel-based face detecion and tracking with adaptive control by kalman filtering,” in Proceedings of the IEEE International Worshop on Intelligent Data Acquisition and Advanced Systems: Tecnology and Applications, pp. 434-439, Italy, September 2009.