A ROBUST AND AUTOMATIC FACE TRACKER DEDICATED TO BROADCAST VIDEOS

Elise Arnaud [1], *Brigitte Fauvet* [2], *Etienne Mémin* [1], *Patrick Bouthemy* [2]

[1] IRISA / Université de Rennes 1 [2] IRISA/INRIA
Campus universitaire de Beaulieu, 35042 Rennes cedex, France

ABSTRACT

Because of their lack of rules, general broadcast videos are more difficult to analyze than news or sport videos. To retrieve human interventions in this context, a robust face tracker is needed. The approach we investigate for face tracking combines three main modules that are a face detector, a region-based tracker and an eye tracker. The region-based tracker relies on a robust parametric motion estimation technique. The eye tracker is based on a Kalman filter. The analysis of the coherence of the trackers output provides an efficient way to detect profile positions and tracking errors. We have thus defined an entirely automatic tracker, able to manage several appearing/disappearing faces, without any *a priori* knowledge on the image sequence. Experimental results on broadcast videos demonstrate its efficiency to deal with large and rapid motions, occlusions and faces in profile position.

1. INTRODUCTION

Human face detection and recognition is an essential component in content-based video indexing and retrieval; especially in the audiovisual domain where a lot of news, documentaries or movies are broadcast and stored. Usual important applications are concerned with retrieving different TV interventions (interviews, shows, etc.) of a given person in a video or a large collection of TV broadcast videos. In this context, a complete chain of video processing modules required to allow human face retrieval should involved a face detector, a face tracker, a clustering face module and a recognition face module. In this paper, we will focus on the face tracking algorithm applied to broadcast videos. It has to provide the longest video segments displaying the person(s) of interest in order to improve the performance of the other modules (face recognition or face clustering process). The challenge is difficult since, in the video documents to be processed, there is no restriction on the viewed scene or on the face characteristics (skin color, expression, pose, etc.). Indeed, one has to deal with the following issues:

- several faces could appear or disappear in a shot and these events have to be properly managed,
- size and pose of the faces can not be *a priori* fixed,
- illumination conditions could vary depending on the nature of the video: shows, movies, etc.

Related work Focusing on methods that can handle several faces in a video sequence, several approaches have been investigated to track human faces. For each tracked face, three steps are involved that are initialization, tracking and a stopping procedure.

Most of the developed methods use a face detector as the initialization of their tracking process [2] [3] [4]. The choice of the most adapted features to track is the difficult key point. The exploitation of skin color is a common choice in order to be invariant to scale or orientation changes [2] [3] [5]. However, such solutions often depend on a learning set dedicated to the type of processed videos or on an update procedure of the color histogram describing the region of interest. They are not guaranteed to be easily expendable to unknown videos with different face types or varying illumination conditions. On the other hand, some trackers are based on the study of facial features (eyes, nose, mouth) [6] [7], or motion information [5]. Obviously, different cues may be combined. Depending on the chosen features describing the face, the estimation of scale and pose changes may be achieved using a template matching procedure [3], a stochastic filter (as Kalman filter [7] or particle filter [4] [5]) or a motion estimation technique [6] [2]. Finally, the stopping procedure is rarely discussed. This constitutes a major deficiency of face tracking algorithms that are generally not able to stop a face track in case of tracking error.

Proposed approach The face tracker we propose exploits global motion information associated to the face region, and local motion information associated to the eyes of the tracked face. Our approach combines three main modules that are: (a) a face detection procedure, (b) a region tracker and (c) an eye tracker. The principle of our algorithm can roughly be described as follows. Initialized by the face detector, each image region of interest is tracked using a region-based method. This tracker relies on a robust parametric motion estimation technique. To detect and cope with drifts - that may occur in case of tracking faces that change from a front-view position to a profile position - an eye tracker is defined. The use of the latter module allows us to check the region-based method results and significantly enhances the tracker robustness. The analysis of the coherence of the two trackers output supplies a practical stopping criteria, as well as information on tracked face positions (front-view or profile). This original approach leads to an efficient algorithm, robust to complex situations such as fast head motion, face in profile or occlusion and is able to deal with several appearing/disappearing faces.

This paper is organized as follows. We first briefly describe the face detector we use in Section 2. Then, the module combining the region-based tracker and the eye tracker is detailed in Section 3. The analysis of the coherence of the two trackers output is presented in Section 4. Experimental results on broadcast video sequences are reported in Section 5 to demonstrate the tracker efficiency and accuracy. Concluding remarks are given in Section 6.

2. FACE DETECTOR

Detecting faces in an image is a difficult problem due to the large variability of face possible appearances. We use the method developed by Garcia and Delakis [8]. It consists in a fast filtering technique corresponding to successive convolution and sub-sampling operations. The involved parameters are determined by neural learning from a large database of examples. This method is both real-time and robust to scale, pose and lighting conditions. Its output is a list of regions. Nevertheless, some mis-detections may occur, in particular facing faces in profile position. Such failures make a tracking method only based on successive matching of the detection results along the sequence inefficient.

The face detection module is performed regularly along the image sequence. For each new detected face, the associated region is used as the initialization of a new track. Moreover, for each detected face, an eye localization step is applied. It is achieved with an eigenfeature method that is derived from an eigenface detection approach [9]. As it will be explained in the following section, we exploit all the available face and eye detections in our eye tracker module.

3. ROBUST TRACKING: ASSOCIATION OF A FACE REGION TRACKER AND AN EYE TRACKER

3.1. Face region tracking

As mentioned in the previous section, a tracker only built on successive matchings of face detection results along the sequence is inefficient. In order to cope with detection failures, we propose to rely on a motion estimation technique. In the first image, the image regions supplied by the face detection module are used to initialize the tracks associated to each character of the scene. The tracking of each of these regions is governed by a robust parametric motion estimation technique [10]. The latter reliably computes a 2D parametric model representing the dominant image motion within the considered support $\mathcal{R}$. The motion vector of a point $\mathbf{s} \in \mathcal{R}$ between the images I_{k-1} and I_k is modeled as a polynomial function of the point coordinates:

$$\mathbf{u}_k(\mathbf{s}) = P(\mathbf{s})\ \boldsymbol{\theta}_k$$

where $\mathbf{u}_k(\mathbf{s})$ denotes the estimated motion vector of pixel $\mathbf{s} = (x, y)^t$ and $\boldsymbol{\theta}_k$ the parameter vector which contains the polynomial's coefficients. $P(\mathbf{s})$ is a matrix related to the chosen parametric model whose entries depend on the spatial coordinates x and y. The parameter vector $\boldsymbol{\theta}_k$ is estimated through the minimization of a robust function:

$$\boldsymbol{\theta}_k = \arg\min_{\boldsymbol{\theta}} \int_{\mathcal{R}} \rho \left(\boldsymbol{\nabla} I_k^t(\mathbf{s}) P(\mathbf{s}) \boldsymbol{\theta} + \frac{\partial I_k}{\partial t}(\mathbf{s}) \right) d\mathbf{s},$$

where ρ is a robust cost function allowing to cope with outliers (occlusion area, secondary motions, etc...). The minimization is achieved through a Gauss-Newton-type multiresolution procedure, which allows handling large magnitude motion. Choosing an affine motion model and defining the support $\mathcal{R}$ as the face region itself provides a robust way of finding the region of interest in the following image.

Along the video shot, additional characters may appear. The criterion to add a track is linked to the face detection module. A matching step (based on overlapping criterion between the detected regions and the already tracked face regions) is performed in order to associate a detected region to an already existing track. A non associated region is then considered as an initialization for a new track. Let us point at that the automatic handling of new entering faces constitutes a important advantage of our approach.

This region tracking procedure is efficient as long as the parametric motion estimation is relevant. Nevertheless, motion estimation can degrade in case of too fast head motions, and especially in case of changing faces from front-view position to profile position. As a consequence, the tracking may diverge. We propose to add an eye tracker module to detect and overcome these problems.

3.2. Eye tracking

For a given detected face, we obtain the localization of the two eyes with the technique described in [9]. The two eyes are tracked separately with an algorithm based on the point tracker developed in [11]. This method relies on a Kalman filter and allows us to cope efficiently with occlusions and abrupt changes. The specificity of the approach we have developed for eye tracking is to consider an *a priori*-free system. The state of the system at time k, $\mathbf{x}_k$, is chosen as being the location of the eye in I_k. The used system combines a dynamic model relying on an instantaneous motion vector and measurements provided by a matching technique[1]:

[1] Let us note that when both dynamic and measurement equations depend on the image data, the use of the Kalman filter can be justified through a peculiar estimator named conditional minimum variance estimator [11]

- The dynamic model is specified on-line thanks to the previously described motion estimation technique that allows us to compute an instantaneous motion vector of the eye $\mathbf{u}_k(\mathbf{x}_k)$. The linear dynamic equation is given by:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{u}_k(\mathbf{x}_k) + \mathbf{v}_k = \mathbf{x}_k + P(\mathbf{x}_k)\,\boldsymbol{\theta}_k + \mathbf{v}_k, \quad (1)$$

where $\mathbf{v}_k$ is a zero-mean Gaussian white noise of covariance $\mathbf{Q}_k$. Let us denote σ_k the ratio between the number of outliers and the number of inliers involved in the estimation of $\boldsymbol{\theta}_k$. The analysis of σ_k along the tracking process can be exploited as a motion estimation quality and gives us a criterion to fix the dynamic noise covariance $\mathbf{Q}_k$: a small value if the ratio is close to 1 or a large one if the ratio decreases. The switch decision depends on the increasing/decreasing jump of σ_k supplied by a statistical text [1](Page-Hinkley test).

- The considered measurement is provided by a matching step and corresponds to correlation peak w.r.t. a reference point. A reference point is chosen as being the detected eye location in the last available image detection. The considered similarity function is a sum-of-square-difference criterion. Denoting $\mathbf{z}_k$ the observation at time k, we have the following linear measurement equation:

$$\mathbf{z}_k = \mathbf{x}_k + \mathbf{w}_k, \quad (2)$$

where $\mathbf{w}_k$ is a zero-mean Gaussian white noise of covariance $\mathbf{R}_k$. $\mathbf{R}_k$ is estimated on-line by modeling the correlation surface as a probability distribution of the true match location. This estimation allows the tracker to be robust to occlusions. More details on this stage may be found in [11].

In case of a face in a profile position, only one eye is correctly tracked, since the second one is occluded. However, the tracking of the latter is continued, in order to recover the right eye location as soon as the character of interest will be in a front view position and a new eyes detection is available.

4. ANALYSIS OF THE TRACKERS COHERENCE

As the face region and eyes have been tracked separately, it is appropriate to analyze the coherence of their output. In the validation step of the global tracking process, the coherence of the estimated face and eye positions is used in order to detect any track drift or faces in profile positions. Several cases may be encountered:

- **Both eyes are in the face region.** This case corresponds to an efficient tracking process, when the outputs of the two modules are coherent.

- **Only one eye is situated in the face region.** A profile position may be described by one eye correctly tracked and positioned in the face region and the other one completely wrongly tracked (out of the face region). In this case, the tracking is continued. Let us point out that the ability of our method to handle profile positions constitutes one of the major advantages of our approach.

- **None of the two eyes are in the face region.** Incoherence between the two trackers is relevant to a tracking drift. In that case, a delay of ten images is introduced. It allows us to recover a track temporally disturbed by an occlusion. If after this delay, both eyes are still out of the face region, the track is stopped.

For a given face, the overall tracking method is summarized in Fig. 1. It is initialized by the face detector, and involves the combination of a face region tracker and an eye tracker. The eye tracker use eyes detection that may be regularly available. Then, the analysis of the respective coherence of the updated face region and eyes location allows us to validate or not the associated track.

5. EXPERIMENTAL RESULTS

In order to demonstrate the accuracy and efficiency of our method, experiments have been carried out on long broadcast videos and on an opera video. The latter involves frequent fast motions and profile positions of the main actors. In the reported results, the rectangle represents the tracked face region and the crosses indicate the tracked eyes.

We first comment results obtained on a shot belonging to the opera video. This shot contains large and rapid movements of the actress filmed in a profile position. The results are displayed on Fig. 2. Whereas only one face detection is available, it can be noticed that the face trajectory is successfully recovered over the shot. The second result (Fig. 3), corresponding to another shot of the opera video, demonstrates the robustness of our tracking method to occlusions.

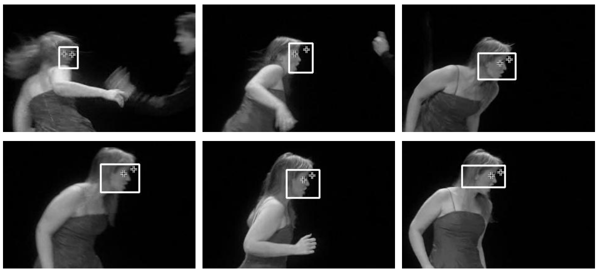

Fig. 2. Results of profile head tracking with only one face detection input used as initialization.

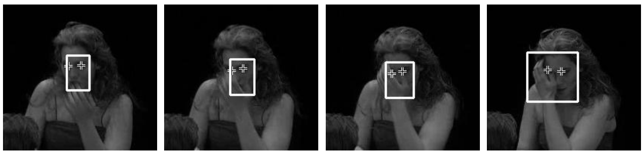

Fig. 3. Results of the tracking process in case of occlusion of the face by the hand.

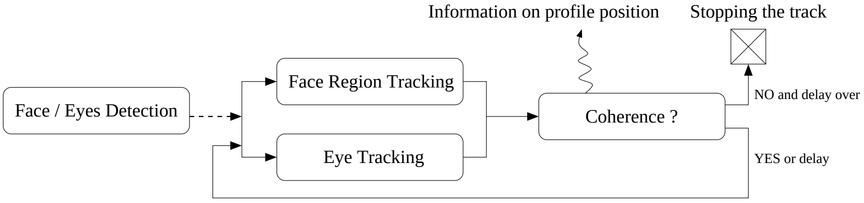

Fig. 1. Overall tracking algorithm for a given face

Table 1. Tracking results obtained on a 2 hours broadcast video

Characters	(a) number of TV interventions	(b) average length of TV intervention	(c) average percentage of recovered TV interventions	(d) average number of tracks per TV intervention
Character 1 (interview)	13	193	96%	1.2
Character 2 (interview)	15	253	99%	1
Character 3 (show)	8	910	91%	1.2
Character 4 (presenter)	73	289	92%	1.1

The last reported experiment is intended to show the ability of our method to automatically track faces in image sequences without any *a priori* knowledge and in general situations. To that end, we have run our tracking method on a 2 hours TV broadcast video corresponding to an entertainment TV program, where invited people are presenting their shows and are interviewed. The four main characters present in the video have been studied. The results of our algorithm has been compared to a ground truth in order to evaluate its performance. The ground truth has been collected by manually detecting all the TV interventions of these characters in the TV program. The obtained results are given in Table 1. For each considered character, we indicate (a) the total number of TV interventions in the video ; (b) the average length of the TV interventions in terms of number of images ; (c) the average percentage (over all the TV interventions of the considered character) of images where the character has been effectively tracked ; and (d) the average number of tracks computed per TV intervention (it should equal to one in the best case).

As shown in Table 1, the processed video contains a large number of long tracks (more than 100 segments). The high percentage of correctly tracked faces (more than 90 %) reveals that our tracker is able to cope with challenging issues such as head pose changes, large and rapid motions, occlusions, or changes of scale. Moreover, it usually delivers only one track per shot (at most, two tracks).

6. CONCLUSION

We have described a robust and completely automatic face tracker dedicated to general TV broadcast videos. Our method combines a region-based tracker and an eye tracker, providing a simple and efficient way of verifying the tracking results. It has been successfully applied to broadcast videos of several hours and the problem of appearing/disappearing faces has been automatically handled by our tracking method. Future work will be concerned with the design of a face recognition method exploiting the tracker output.

Acknowledgements: This work was supported by the French Ministry of Industry within the RIAM FERIA project. The videos were provided by ARTE and INA.

7. REFERENCES

[1] P. Bouthemy, M. Gelgon and F. Ganansia "A unified approach to shot change detection and camera motion characterization," in *IEEE Trans. on Circuits and Systems for Video Technology*, 1999, pp. 1030–1044.

[2] P. Gejguš and M. Šperka, "Face tracking in color video sequences," in *Conf. on Computer Graphics*, 2003.

[3] Z. Liu and Y. Wang, "Face detection and tracking in video using dynamic programming," in *ICIP*, 2000.

[4] K. Mikolajczyk, R. Choudhury, and C. Schmid, "Face detection in a video sequence - a temporal approach," in *Proc. CVPR, Kauai, Hawaii, USA*, Dec. 2001.

[5] H. Wu and J.S. Zelek, "The extension of statistical face detection to face tracking," in *CVPR Workshop on Face Processing in Video, Washington*, 2004.

[6] A.W. Senior, "Recognizing faces in broadcast video," in *Int. Workshop on Recognition, Analysis and Tracking of Faces and Gestures in Real-Time Systems*, 1999.

[7] X. Wei, Z. Zhu, L. Yin, and Q. Ji, "A real time face tracking and animation system," in *CVPR Workshop on Face Processing in Video, Washington*, 2004.

[8] C. Garcia and M. Delakis, "Convolutional face finder: A neural architecture for fast and robust face detection," *IEEE Trans. PAMI*, vol. 26, no. 11, 2004.

[9] B. Moghaddam and A. Pentland, "Probabilistic visual learning for object representation," *IEEE Trans. PAMI*, vol. 19, no. 7, 1997.

[10] J.-M. Odobez and P. Bouthemy, "Robust multiresolution estimation of parametric motion models," *Journal of Visual Communication and Image Representation*, vol. 6, no. 4, pp. 348–365, 1995.

[11] E. Arnaud, E. Mémin, and B. Cernuschi-Fríias, "Conditional filters for image sequence based tracking - application to point tracking," *IEEE Trans. IP*, january 2005.