SAP: A robust approach to track objects in video streams with Snakes And Points

Valerie Gouet & Bruno Lameyre
CEDRIC/CNAM - 292, rue Saint-Martin - F75141 Paris Cedex 03
valerie.gouet@cnam.fr , bruno.lameyre@free.fr

Abstract

This paper presents a robust and generic approach of object tracking in video sequences. Here, the object to track is described by considering two well-known image primitives: first, its content is described with *Points of interest*. Such points are automatically extracted and then characterized according to a selective spatial appearance-based model. Second, the object envelope is described with a *Snake*. The originality of the SAP approach consists in a complementary use of these two primitives: the snake allows to reduce the points tracking to a limited area in each frame, and the spatial point description is exploited during the snake tracking, making the process robust to wide occlusions. Since no model of trajectory is considered, the approach is robust to wide motions of object and camera. The relevance of this approach has been evaluated on several video streams. Results obtained with the most representative of them are presented in this paper. The algorithms involved have been implemented with the aim of achieving near real-time performance.

1 Introduction

In a variety of applications of image technology, such as medical image analysis, video surveillance or scene monitoring, it is desirable to track objects in video sequences. Considerable work has been done during the past few years in object tracking. There is no theory for the segmentation of moving objects in videos, the methods depend upon the target application. When a model of the moving object does not exist, the encountered approaches usually focus either on image spatial structures, or on temporal tracking with trajectory estimation, or on both. Different kinds of approaches exist, they are usually based on region segmentation [20, 13], blobs [16], color histograms [21], optical flow [4], point [25] or snakes [5] tracking.

The paper is organized as follows: section 2 describes the approach of object content description we investigate, which is based on points of interest. In section 3, we remind of snakes principles and we present a novel approach of object segmentation and tracking combining snakes and points of interest. Experiments on video streams are presented in section 4 to highlight the contributions of our method, before concluding in section 5.

2 Object tracking with points of interest

Points of interest are involved in many applications, like stereovision, image retrieval or scene monitoring. For all of these applications, the extracted points usually represent sites where the information is considered as perceptually relevant. Ideally, the point detector

BMVC 2004 doi:10.5244/C.18.76

should be able to repeat the extracted points from an image to another whatever the photometric/geometric transformations involved. Many point extractors have been proposed, see for example the comparison study [24]. The most popular one is probably the Harris and Stephens detector [11] with its adaptations [23, 19, 17, 15].

Temporal approaches of feature point tracking exist for *point trajectory estimation*. Classically, the encountered techniques involve a function cost defined for three consecutive frames. Different linking strategies are applied to find the correspondences and optimize the trajectories. The first solution is the one developed by Sethi and Jain in 1987 [25] and called Greedy Exchange algorithm (GE). This algorithm is based on a cost function which penalizes the changes of direction and the magnitude of the speed vector. Salari and Sethi [22] solve the missing and spurious measurements problem of the GE approach by introducing phantom points. In [26], a modified version of the GE algorithm is proposed for point trajectory estimation in sequences of facial images. In [6], the algorithm "IPAN tracker" described is based on the idea of competing trajectories. The paper also presents a performance evaluation of feature points tracking approaches.

Most of the approaches listed above estimate a trajectory according to a local model of trajectory. They do not exploit the visual appearance of the points to track. Since they involve a model of trajectory, their main drawback is to be not robust to wide deformation of non-rigid object and to wide occlusions. In this paper, we focus on spatial appearance-based tracking approaches. Such techniques do not impose any constraint on the trajectory of the point to track and may allow wide occlusions, as it will be demonstrated.

Traditional approaches involving a *spatial description of points of interest* come from stereovision or more recently from image retrieval applications. From the works of Koenderink [14] and Florack [7] on the properties of local derivatives, a lot of work has been done on differential descriptors [23, 1, 17, 9]. The point signature is based on a combination of differential quantities, computed on grey value or color images. They can be adapted to obtain invariance under translation, rotation, changes of scale, affine or illumination transformations according to the case. Usually employed in the context of texture classification, frequency approaches have been developed by considering the Gabor transform that allows to take spatial relations between points into account [27]. A recent performance evaluation of local descriptors [18] has shown that the local descriptor SIFT proposed by Lowe [15]for object recognition performs best.

2.1 Our approach of point of interest tracking

We did not make the choice of employing the SIFT descriptor in our prototype, first because it involves a high dimensional features set (128 items for each keypoint [15]), making it not applicable for real-time video tracking purposes. Second, this descriptor is invariant to several image transformations, making it efficient for object recognition but not optimal for video streams where consecutive frames differ by small transformations (this idea will be developed in section 2.2). Therefore, the characterization employed here is the local jet of the signal which approximates the point neighborhood by a set of image derivatives and which is invariant to image translation. Up to order n, it can be expressed for the point (x,y) as follows:

$$J(x,y,\sigma) = \{I_{i_1 \ldots i_k}(x,y,\sigma)/k = 0,..n\} \quad (1)$$

where $I_{i_1 \ldots i_k}(x,y,\sigma)$ represents the k^{th} image derivative relative to the $i_1 \ldots i_k$ variables (x and y) and σ the size of the Gaussian smoothing applied during the derivatives computation. Under the gaussian assumption, the similarity measure traditionally combined with

this characterization is the Mahalanobis distance $\delta^2(v_1,v_2) = (v_1 - v_2)^T \Lambda^{-1}(v_1 - v_2)$. $v_i \in V$ with $V \subset \mathbb{R}^d$ is the feature space associated to the chosen characterization. The involved covariance matrix Λ takes the different magnitudes, possible correlations and variability of the feature components into account. In the rest of the paper, such a point characterization space will be noted (V_d, δ^2).

Point matching algorithm. A specific model of trajectory is not exploited here. We only suppose that the point p_i^j characterizing the object O_i of frame F_i has its corresponding point in frame F_{i+1} inside an area which is simply modelled by a circular window W_t of size t centered on p_i^j. The matching algorithm consists in finding in (V_d, δ^2) the nearest neighbor p_{i+1}^k of p_i^j, with p_{i+1}^k in $W_t(p_i^j)$. A match having a distance δ^2 higher than a given threshold is eliminated. Under some hypothesis, the threshold can be automatically chosen from the χ^2 table. Then a classical cross-matching algorithm is applied in (V_d, δ^2) in order to build a set of matches $\{(p_i^j, p_{i+1}^k)\}$ with points involved in each match at best one time. Semi-local geometric constraints that consider spatial relations between neighbor points can be added to enrich the point matching algorithm [23, 9].

Our algorithm privileges the visual similarity of points of interest. The t parameter can be viewed as a function of the velocity of the point to track. It can be estimated from the couple (p_{i-1}^l, p_i^j) of matched points, as in the approaches of point trajectory estimation. In that case, the matches involve points which are visually similar and constrained by a particular velocity from frames F_{i-1} to F_{i+1}.

2.2 A study on models of noise

Point descriptors are subject to different kinds of noises: by definition, they are at best only quasi-invariant [2] to any point of view and in practice, they are sensitive to image acquisition (sensors and sampling errors may be important for images coming from video sequences), to numerical errors, to points of interest delocalization, etc.

These considerations show the importance of the similarity measure which must be carefully chosen for the considered descriptor to achieve best performances. An optimal similarity measure is directly related to the shape of their variability. When considering the Mahalanobis distance, this noise can be integrated in the similarity measure via the covariance matrix Λ. When a model of noise of the components cannot be specified, the way to estimate Λ comes down to different empiric solutions:

- Estimating Λ from all the available data. This simple solution generates weights that are not discriminant, since representing a rough model of noise. Even so, this is the most common solution encountered to compare features with the Mahalanobis distance. In the rest of the article, the similarity measure obtained from such an estimation will be noted δ^2_{rough};
- Estimating Λ from points of interest whose local neighborhood is submitted to synthetic photometric and geometric transformations and perturbations that usually apply to images;
- Estimating Λ from training sequences of real images. Several points on different images with representative perturbations are tracked and a combination of the covariance matrices obtained can be used as the model of noise of point characterization. This solution has been adopted in [23] for image retrieval. The Mahalanobis distance obtained from such an estimation will be noted $\delta^2_{trained}$.

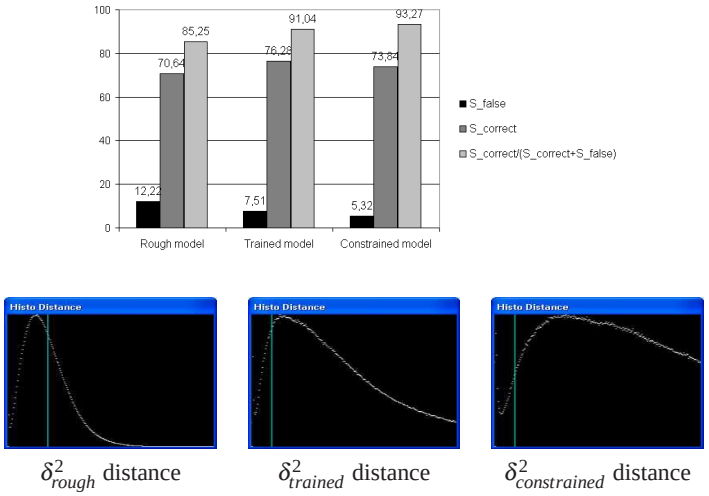
The point characterization approach used for our first experiment is the local jet up to order 2, implying the feature space (V_6, δ^2). We evaluated the two models of variability δ^2_{rough} and $\delta^2_{trained}$ through our point tracking algorithm. Several training video sequences involving various contents and image transformations were considered to estimate the covariance matrices. The Λ_{rough} one was computed from the extracted points in all the sequences. The $\Lambda_{trained}$ one was estimated from points tracked in several frames. The training sequences were calibrated in order to automatically determine the sequences of extracted points and to evaluate the efficiency of our tracking algorithm: the camera was static and the models of object motion to track were known.

If we consider a video containing N frames $F_{i\ \forall i \in [1..N]}$ with n_i points of interest extracted on the object to track O_i in frame F_i, we compute two kinds of scores:

$$S_{correct} = \frac{1}{N-1} \sum_{i=1}^{N-1} \frac{|\{CM(O_i, O_{i+1})\}|}{\min(n_i, n_{i+1})} \qquad S_{false} = \frac{1}{N-1} \sum_{i=1}^{N-1} \frac{|\{FM(O_i, O_{i+1})\}|}{\min(n_i, n_{i+1})} \tag{2}$$

$S_{correct}$ involves the number of correct matches $\{CM(O_i, O_{i+1})\}$ obtained between O_i and O_{i+1}, whereas S_{false} is related to the false ones $\{FM(O_i, O_{i+1})\}$. These quantities are normalized according to the number $M_{i,j}$ of effective matches existing between O_i and O_{i+1}. For simplicity, $M_{i,j}$ is replaced by its upper boundary $\min(n_i, n_{i+1})$. A match is considered as correct if the involved points respect the motion of the object (which is known for the evaluation), and false if not.

Figure 1 presents the point matching scores and histograms of distances obtained according to three models of variability. For the moment, we only focus on the Λ_{rough} and $\Lambda_{trained}$ ones.



δ^2_{rough} distance $\delta^2_{trained}$ distance $\delta^2_{constrained}$ distance

Figure 1: *Matching scores and histograms of distances obtained with Λ_{rough}, $\Lambda_{trained}$ and $\Lambda_{constrained}$ models. The third score presented is the ratio of correct matches according to the whole set of matches found. The associated χ^2 threshold is represented by a vertical line on the histograms ($\chi^2 = 12.6$ for $d = 6$).*

As expected, the scores obtained clearly confirm that a model of noise is necessary to exhibit the relevance of the point characterization employed. The Λ_{rough} model does not represent an efficient model of the variability for the point characterization. Many measures δ^2_{rough} produced are smaller than the associated χ^2 threshold, leading to high

rates of false matches. The $\Lambda_{trained}$ model provides a more selective measure $\delta^2_{trained}$. Note that the peak of the corresponding histogram is moved behind the threshold. The number of correct matches is better while the number of false matches is reduced, making 6% better the ratio of correct matches compared to the matches found.

On the choice of the point characterization. The scores presented in figure 2 remind the importance of the choice of the point characterization. Here, two characterizations are tested: the local jet of equation 1 which is invariant to translation and the Hilbert's differential invariants[1] which combine local jet items to achieve invariance to image rotation. The model of variability employed here is the $\Lambda_{trained}$ one. The covariance matrix has been trained on sequences involving objects moving according to an image rotation. The sequence used for computing the scores contains the same motion but does not belong to the training sequences.

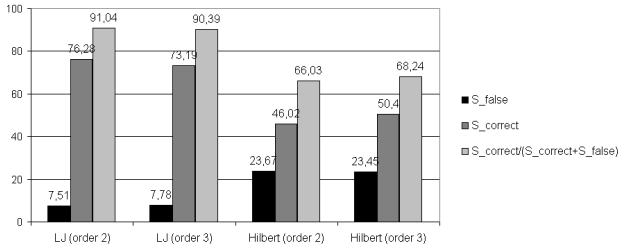


Figure 2: *Matching scores obtained according to the $\Lambda_{trained}$ model, with two point characterizations: the local jet and the Hilbert's differential invariants (up to order 2 and up to order 3).*

The best matching results are obtained with the local jet, despite the fact that this descriptor is not invariant to rotation, as the Hilbert's one. This observation leads to the following conclusion we generalize to other usual image transformations: when images differ by small transformations (it is typical in video sequences), the way to develop the most robust descriptor is to keep it the less invariant possible and to learn the variability generated by the small transformations involved through the similarity measure. Developing features invariant to several transformations naturally leads to a less selective description.

Another observation concerns the sensitivity of high order derivatives. Here, adding order 3 does not improve the scores significantly. These precise features do not resist to the poor quality of video images, as demonstrated in [10] for image retrieval.

Constraining invariance. Estimating the features variability on training sequences also allows to specify more precisely the range where there are allowed to vary, without to enforce complete invariance as it is usually performed. For instance, it is possible to constrain invariance only in a range of rotation angles. Indeed, in video sequences, it is rare that two consecutive frames (or parts of frames) differ by large rotation angles. To confirm this idea, the $\Lambda_{trained}$ model was re-estimated on training sequences differing from small transformations; the model obtained is $\Lambda_{constrained}$. We considered image rotations with angles smaller than 22°. Matching results involving such a model are also presented in figure 1. The best results are obtained with this model: the numbers of false and correct

[1] For grey value images, the local jet involves 6 features up to order 2 and 10 features up to order 3, while the Hilbert's differential invariants involve 5 features up to order 2 and 9 features up to order 3.

matches slightly decrease, but the ratio of correct matches gains 2%. As illustrated with the corresponding histogram, the similarity measure $\delta^2_{constrained}$ is even more selective.

3 Object segmentation and tracking with snakes

The Snakes theory was born in 1987 with the work of Kass et al. [12]. A complete state of art about snakes can be found in [3]. Snakes are widely used for segmentation, shape modelling and motion tracking. A snake can be represented as a parametric curve $\mathscr{C} : v(s) = (x(s), y(s)), \forall s \in [0..1]$. From a given starting position, the snake deforms itself in order to stick to the nearest salient contour. The snake behavior and its evolution are governed by a weighted combination of internals and externals forces and is computed as an energy function E to minimize, with $E = \int_0^1 (\alpha(s)|v'(s)|^2 + \beta(s)|v''(s)|^2)ds - \int_0^1 |\nabla I(v(s))|^2 ds$. Minimizing E is not easy. A numerical solution consists in considering a discrete representation of $\mathscr{C}$ and in developing an algorithm which proceeds iteratively.

Regularization of the curve. The discrete snake is a vector of node $(1..i)$ linked by segments. Three forces are usually applied on each node of the snake. The first one is a *stretching* force which can be written as $E_{stretching} = |\sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2}|^n - D_{ref}$ where D_{ref} represents the initial distance between two consecutive nodes. The second force is the *bending* force which can be written as $E_{bending} = (x_{i-1} - 2x_i + x_{i+1})^2 + (y_{i-1} - 2y_i + y_{i+1})^2$. This curvature is an approximation which is faster to compute. The third force is the *external* force which comes from the image itself: $E_{external} = -|\nabla I(x,y)|^2$. Some temporal forces can be added to help during the tracking [5].

Minimizing the energy. In order to reduce computation time, a determinist algorithm which reduces the total energy of the snake by reducing the energy of each node separately was chosen. This process is iteratively repeated as the snake energy decreases. During the optimization, the candidate neighbors of each node stay on a segment orthogonal to the tangent of the node. For each candidate, the energy is computed as described above.

3.1 Exploiting points of interest to enhance snake tracking

At present, we are able to characterize the view O_i of an object in a frame F_i with a set of interest points noted P_i and with a discrete snake noted S_i. The complete object characterization obtained is the couple (P_i, S_i). In this section, we propose a method consisting in exploiting the P_i features to make the snake tracking more robust.

Let consider two sets of points P_i and P_j characterizing two views O_i and O_j of the same object in two frames F_i and F_j. Matching these two sets (or subsets) allows to estimate the image transformation $T_{i,j}$ existing between O_i and O_j. Here, the objective is not to estimate the motion of the object between F_i and F_j, but only the object evolution in the frames. Consequently, $T_{i,j}$ can be used to enhance the snake tracking between two consecutive frames F_i and F_{i+1}: the snake S_{i+1} can be initialized with $T_{i,j}(S_i)$, before optimizing it for the view O_{i+1}.

Robustness against wide occlusions. According to the object characterization we have adopted, we consider that an object becomes occulted in a frame F_i when few points P_i can be matched with P_{i-1}. In such a case, points are extracted in the whole frames $F_{j,j \geqslant i}$ as long as the object is occulted. The corresponding sets obtained are called $P_{j,global}$.

Now, let suppose that we have at our disposal the description noted $(P_{i_{ref}}, S_{i_{ref}})$ of one of the views $O_{i_{ref}}$ before the object occlusion, and the set $P_{j,global}$ extracted from the frame F_j when it reappears. $P_{i_{ref}}$ and $P_{j,global}$ can be compared according to the approach detailed in section 2.1. It is reasonable to suppose that the points of $P_{j,global}$ which are involved in the matches obtained give a characterization P_j of the view O_j. Then estimating $T_{i_{ref},j}$ from some of the points $(P_{i_{ref}}, P_j)$ in correspondence allows to initialize in F_j a snake with $T_{i_{ref},j}(S_{i_{ref}})$. This technique supposes that a view $O_{i_{ref}}$ which is quite similar to O_j exists and that $(P_{i_{ref}}, S_{i_{ref}})$ is available. To do that, our approach consists in storing during the tracking sub-samples $(P_i, S_i)_{i=k_1,..,k_D}$ of the object characterization in a FIFO list called H_D. Under this hypothesis, $(P_{i_{ref}}, S_{i_{ref}})$ can be chosen within H_D as the description which fits better a subset of $P_{j,global}$. The algorithm is illustrated in figure 3.

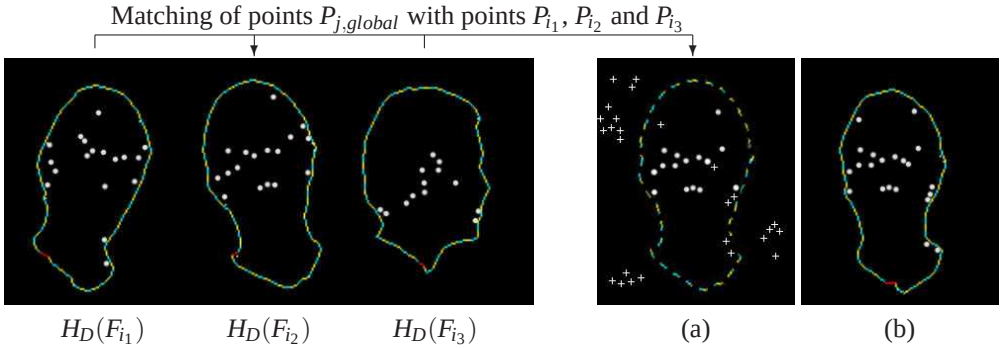


Figure 3: *Tracking of a face after a full occlusion. On the left, 3 items of the history list. On the right, (a) shows the points $P_{j,global}$ extracted on a whole frame F_j after the occlusion. • points refers to the P_j which better match with points of H_D (here with P_{i_2}) and + points are unmatched points. The dotted snake drawn is $T_{i_2,j}(S_{i_2}) = S_{j,init}$. (b) shows the P_j points plus the optimized snake S_j obtained from $S_{j,init}$.*

3.2 The complete algorithm of tracking with snakes and points

The SAP algorithm proceeds as described in Algorithm 1. Points are extracted using the Precise Harris detector and characterized with the local jet associated to $(V_6, \delta^2_{trained})$. Two points sets are matched according to the matching approach presented in section 2.1.

In this algorithm, the window $W_{S_{i-1}}$ considered for the points of interest extraction in frame F_i is based on the snake computed in frame F_{i-1}. The area associated to S_{i-1} only gives in F_i a first approximation of the area where to extract the points that will characterize the object. Since the points to track may have moved between the two frames, it is necessary to consider an enlarged surface. We consider a simple dilatation of the surface associated to S_{i-1}, noted $\mathscr{A}(S_{i-1})$. The size of the dilatation can be viewed as a function of the points velocity, as for the parameter t of the window W_t used during the point matching process between two frames (see section 2.1).

4 Results of object tracking

In this section, we evaluate the robustness of the SAP prototype against a full occlusion. The actual video resolution is QCIF (352×288) and image is acquired in YUV12 format (4:2:0). Only the Y part, containing the grey level information is exploited.

Algorithm 1: Object tracking with snakes and points of interest.

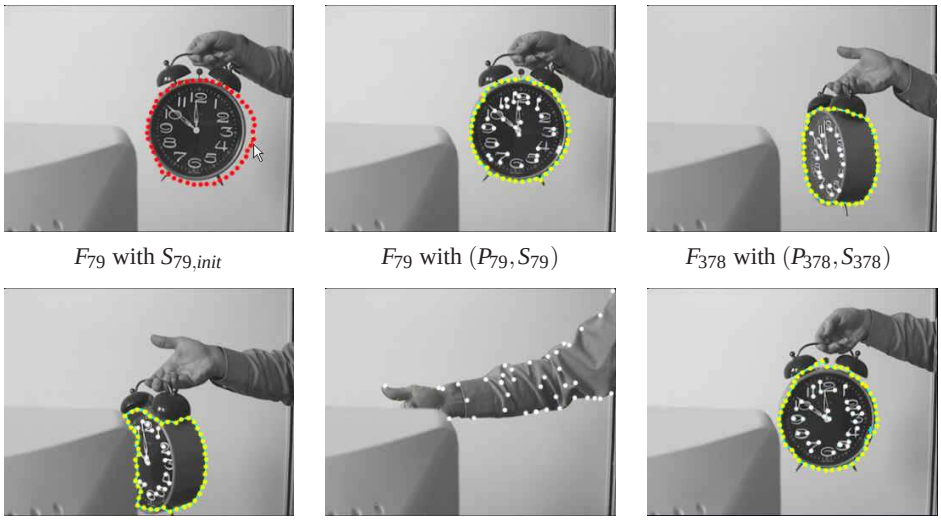
// `Structures initialization`
- Manual surrounding of the object to track in F_1. It gives $S_{1,init}$;
- Optimization of $S_{1,init}$ for the object O_1. A refined snake S_1 is obtained;
- Extraction of a set of points P_1 in F_1 inside the area defined by S_1;

For *each frame* $F_{j,j>1}$ *of the sequence* **do**
  If $(P_{j-1}, S_{j-1}) \neq (\varnothing, \varnothing)$ **then**
    // `The object was globally visible in frame` F_{j-1}
    - Extraction of a set of points P_j in F_j inside an area $\mathscr{A}(S_{j-1})$;
    - Point matching of the P_{j-1} set with the P_j one in (V_d, δ^2);
    - $P_{i_{ref}} \leftarrow P_{j-1}$;
  else
    // `The object was widely occulted in frame` F_{j-1}
    - Extraction of a set of points $P_{j,global}$ in the whole frame F_j;
    - Search in H_D of the P_i set associated with the best score $SCR_H(P_i, P_{j,global})$. It involves a subset $P_j \subset P_{j,global}$;
    - $P_{i_{ref}} \leftarrow P_i$;
  end if
  If *enough* $P_{i_{ref}}$ *points are matched with the* P_j *ones* **then**
    // `The object is globally visible in frame` F_j
    - Estimation of $T_{i_{ref},j}$ from the matches between $P_{i_{ref}}$ and P_j;
    - $S_{j,init} \leftarrow T_{i_{ref},j}(S_{i_{ref}})$;
    - Optimization of $S_{j,init}$ for O_j. A refined snake S_j is obtained;
    - $H_D \leftarrow H_D + (P_j, S_j)$;
  else
    // `The object is widely occulted in frame` F_j
    $P_j \leftarrow \varnothing$; $S_j \leftarrow \varnothing$;
  end if
  $j \leftarrow j+1$;
end for

Here, the object (a clock) completely disappears behind an obstacle. When disappeared, its trajectory does not follow the same one as before the occlusion, making a model of trajectory unusable. Figure 4 presents particular frames before, during and after the occlusion. The object characterizations associated are superimposed on the frames.

About computation time. For such an application, computation time depends on many factors as the video input format and resolution, the frame rate, the number of feature points extracted in each frame, the number of frames in H_D, the area of the targeted object, etc. All the algorithms developed have been chosen to be real-time compatible. At present, the optimization phase have not yet been made but we think that real-time is achievable. The following estimations give an idea of the actual performances, based on an Intel Centrino 1.6 Ghz CPU computer:

- Snake used alone (as object tracker): 25 ms/frame (40 frames/sec);
- Snake used in cooperation with feature points tracker: 80 ms/frame (12 frames/sec);



F_{79} with $S_{79,init}$ | F_{79} with (P_{79}, S_{79}) | F_{378} with (P_{378}, S_{378})

F_{606} with (P_{606}, S_{606}) | F_{693} with $P_{693,global}$ | F_{761} with (P_{761}, S_{761})

Figure 4: *Evolution of the object characterization* (P_i, S_i) *during the tracking, in the presence of a full occlusion. In frame* F_{79}*, the object is manually surrounded in order to define* $S_{79,init}$*. Frames* F_{606}*,* F_{693} *and* F_{761} *have been respectively taken before, during and just after the occlusion.*

- Time to retrieve the best candidate in H_{256}: 200 ms, depending on the number of points inserted in each history item.

5 Conclusions and future work

In this paper, we have presented a novel approach for object tracking in video sequences, named SAP. The object to track is described by considering two generic image primitives: points of interest and snakes. No model of object nor trajectory is used to achieve the tracking. We focused our work on two particular aspects: first, we tried to develop an appearance-based point characterization the most robust possible to the variability that an image coming from a video may contain. Second, we exploited such a characterization to make the snake tracking more robust. The experiments realized on wide occlusions clearly show the relevance of the spatial description of the points we propose, when a temporal one would be lacking.

The SAP prototype represents the foundations of our object tracking approach. Improvements of this work are various. At present, we are studying a model of variability for the point characterization which is learnt during the object tracking. We also plan to enrich the SAP characterization by adding more temporal information, with the aim of developing a complete model in agreement with studies on human vision, like [8].

References

[1] Adam Baumberg. Reliable feature matching across widely separated views. In IEEE *Conference on Computer Vision and Pattern Recognition*, pages 774–781, 2000.

[2] T.O. Binford and T.S. Lewitt. Quasi-invariants : theory and exploitation. In *Proceedings of DARPA Image Understanding Workshop*, pages 819–829, 1993.

[3] A. Blake and M. Isard. *Active Contours*. Springer, 1998.

[4] G. Castellano, J. Boyce, and M. Sandler. Regularized cdwt optical flow applied to moving-target detection in IR imagery. *Machine Vision and Applications*, 11(6):277–288, 2000.

[5] C. Chesnaud, P. Réfrégier, and V. Boulet. Statistical region snake-based segmentation adapted to different physical noise models. IEEE *PAMI*, 21(11):1145–1157, 1999.

[6] D. Chetverikov and J. Veresty. Tracking feature points: A new algorithm. In *In Proc. International Conf. on Pattern Recognition*, pages 1436–1438, 1998.

[7] L.M.J. Florack, B.M ter Haar Romeny, J.J. Koenderink, and M.A. Viergever. General intensity transformations and differential invariants. *Journal of Mathematical Imaging and Vision*, 4(2):171–187, 1994.

[8] S. Gepshtein and M. Kubovy. The emergence of visual objects in space-time. *National Academy of Sciences*, 97(14):8186–8191, 2000.

[9] V. Gouet and N. Boujemaa. Object-based queries using color points of interest. In *IEEE Workshop CBAIVL*, pages 30–36, Kauai, Hawaii, USA, 2001.

[10] V. Gouet and N. Boujemaa. On the robustness of color points of interest for image retrieval. In IEEE *ICIP'2002*, Rochester, New York, USA, September 2002.

[11] C. Harris and M. Stephens. A combined corner and edge detector. *Proceedings of the* 4^{th} *Alvey Vision Conference*, pages 147–151, 1988.

[12] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contours models. *International Journal of Computer Vision*, pages 321–331, 1988.

[13] M. Kim, J.G. Jeon, J.S. Kwak, M.H. Lee, and C. Ahn. Moving object segmentation in video sequences by user interaction and automatic object tracking. *IVC*, 19(5):245–260, April 2001.

[14] J.J. Koenderink and A.J. Van Doorn. Representation of local geometry in the visual system. *Biological Cybernetics*, 55:367–375, 1987.

[15] David G. Lowe. Distinctive image features from scale-invariant keypoints. *Accepted for publication int the International Journal of Computer Vision*, 2004.

[16] R. Megret and J.M. Jolion. Tracking scale-space blobs for video description. IEEE *Multimedia*, 9(2):34–43, 2002.

[17] Krystian Mikolajczyk and Cordelia Schmid. Indexing based on scale invariant interest points. In *International Conference on Computer Vision*, Vancouver, Canada, July 2001.

[18] Krystian Mikolajczyk and Cordelia Schmid. A performance evaluation of local descriptors. *Intl. Computer Vision and Pattern Recognition*, 2003.

[19] P. Montesinos, V. Gouet, and R. Deriche. Differential Invariants for Color Images. In *Proceedings of* 14^{th} *International Conference on Pattern Recognition*, Brisbane, Australia, 1998.

[20] S. Pateux. Tracking of video objects using a backward projection technique. In *Visual Conference on Image Processing*, volume 4067, pages 1107–1114, Australia, 2000.

[21] P. Perez, C. Hue, J. Vermaak, and M. Gangnet. Color-based probabilistic tracking. In *Eur. Conf. on Computer Vision, LNCS 2350*, pages 661–675, Copenhaguen, Denmark, June 2002.

[22] V. Salari and I.K. Sethi. Feature point correspondence in the presence of occlusion. IEEE *Transactions on Pattern Analysis and Machine Intelligence*, 12(1):56–73, January 1990.

[23] C. Schmid and R. Mohr. Local grayvalue invariants for image retrieval. IEEE *Transactions on Pattern Analysis and Machine Intelligence*, 19(5):530–534, May 1997.

[24] C. Schmid, R. Mohr, and Ch. Bauckhage. Evaluation of interest point detectors. *International Journal of Computer Vision*, 37(2):151–172, 2000.

[25] I. K. Sethi and R. Jain. Finding trajectories of feature points in a monocular image sequence. IEEE *Transactions on Pattern Analysis and Machine Intelligence*, 9:56–73, 1987.

[26] C.J. Veenman, E.A. Hendriks, and M.J.T. Reinders. A fast and robust point tracking algorithm. In *International Conference in Images Processing*, 1998.

[27] J.K.M. Vetterli. *Wavelets and Subband Coding*. Prentice Hall, 1995.