Use of an Evaluation and Diagnosis Method To Improve Tracking Performances

B. Georis[1], F. Brémond[1], M. Thonnat[1] and B. Macq[2]
(1) projet ORION, INRIA
route des Lucioles, 2004
BP 93, Sophia-Antipolis Cedex, France
email:benoit.georis,francois.bremond,monique.thonnat@sophia.inria.fr

(2) Communications and Remote Sensing Lab
Université Catholique de Louvain
1348 Louvain-la-Neuve, Belgium
email: macq@tele.ucl.ac.be

ABSTRACT
This paper presents a general framework for analyzing the evaluation of a tracking algorithm in order to improve it. We first propose a classification of the various errors encountered during the motion detection and the tracking process. This classification is done using a comparison between tracking outputs and ground truth. We propose two evaluation algorithms, a global one and a more precise one. Second, we show how to use this classification to diagnose the tracking errors and to find relevant parameters to solve each problem type and to determine criteria to tune these parameters with respect to the scene environment. This technique is applied to the tracker module of a video interpretation platform whose main goal is to recognize human behaviours. Results are presented for several video sequences taken from a static calibrated camera in three different contexts: a bank agency, a metro platform and an office.

KEY WORDS
Human tracking, performance assessment, supervised evaluation

1 Introduction

Tracking has been extensively studied for many years. Various techniques have been explored, both model-based [1], [2] and model-free [3]. Nevertheless, the tracking problem remains unsolved since there are many sources of ambiguities like shadows, illumination changes, over-segmentation and mis-detection. These difficulties need to be handled in order to make the correct matching decision.

In addition, the increasing number of installed surveillance systems need highly efficient tracking algorithms to be able to recognize for example complex human behaviours. These systems are running 24 hours a day in varying conditions. Our goal is to conceive a generic human tracking algorithm which can adapt itself automatically to a scene change.

Algorithm assessment is a first step to design such robust systems. This is the main topic of the PETS workshop [5] and especially the interesting theoretical work on performance evaluation [4]. In this article, we propose a practical evaluation method classifying tracking errors by comparing tracking outputs and ground truth. We then propose a way to improve tracking performances by introducing a second step which diagnoses the evaluation results.

We show an application of this method with the tracker of a video interpretation platform. This platform consists of four main processing stages: 1) motion detection, 2) Frame to frame (F2F) tracking, 3) long term tracking and 4) behaviour recognition. The main idea through the whole processing chain is to incorporate as much knowledge as we can at each level of reasoning. For instance, we use a human model represented by the mean width and height of a person. A detailed description of the complete system can be found in [6].

In existing systems, parameters controlling the tracking algorithm are normally iteratively improved to repair a specific tracking error until they achieve an acceptable compromise between what is expected and what is observed. Our approach improves the tracking algorithm by repairing globally a whole class of tracking error, using extensively both contextual knowledge of the scene environment and knowledge of the tracking algorithm. We claim that there is image independent information (e.g., human model) which can be used to improve the tracking performance, as done in [7]. The proposed method allows us to make the best use of this knowledge. This paper is restricted to scenes captured by a fixed calibrated camera where the moving objects are humans.

The paper is laid out as follows. Section 2 describes the global evaluation algorithm, with a discussion on the type of video inputs and the ground truth acquisition. Section 3 explains how we refined this first evaluation algorithm into a more precise one. Section 4 shows how to analyze this evaluation to improve the tracking. This method is applied on our F2F tracker and first results obtained with several test sequences are presented. Finally, section 5 concludes and indicates future work.

2 Tracking Evaluation

The overall system is represented in figure 1. The evaluation takes as input ground truth and F2F tracking outputs. It produces a classification of the tracking errors. Finally,

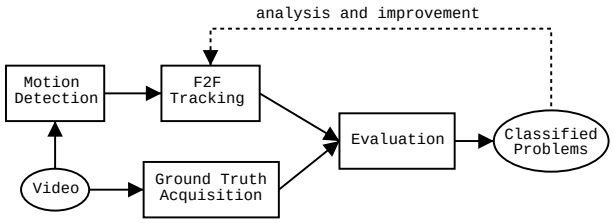

Figure 1. Off-line feedback loop to improve F2F tracking

a careful analysis of these errors enables to improve the tracking.

We concentrate here on the evaluation step. First, we characterize a typical tracking output (section 2.1). Second, we discuss both the influence of the video used for evaluation (section 2.2) and of the ground truth acquisition (section 2.3). Finally, we present the evaluation method.

This is illustrated for a human tracking system but it can be easily generalized.

2.1 Tracking Outputs

Whichever tracking technique, a tracker takes as input at each time t a list of moving regions from the motion detector, and its goal is to produce a temporal graph of moving regions. More precisely, given a set of moving regions $N = \{n_1...n_p\}$ in the current frame and a set of moving regions $O = \{o_1...o_q\}$ in the previous frame, the objective is to create m quantified links between old and new moving regions. The various sequences of edges (links) in this graph represent the various possible trajectories a moving region may have. Moreover, a value can be associated to each link to quantify the confidence about the correspondence between the moving regions connected by the link. The goal of the F2F tracker is to not miss any correspondence and to reduce the ambiguities.

The F2F tracking algorithm is usually guided by two criteria: the first one is the correspondence between the detected moving regions and a human model, and the second one is the temporal continuity indicating that the moving regions detected at time t are expected to be detected at time $t + 1$.

Depending on the motion detector, these moving regions are described by various features which can be simply the 2D size and position. More elaborated techniques can add 3D information or statistical moments, for example. In our implementation, the moving regions are featured by their centre of gravity, their width and their height. All three are defined both in 2D (in the image) and in 3D (in the scene) since we work with calibrated cameras. They are classified according to a semantic class (PERSON, OCCLUDED_PERSON, GROUP_OF_PEOPLE, NOISE or UNKNOWN). Static occlusion is determined using contextual information.

2.2 Video Sequences Selection

The evaluation of a tracking system strongly depends on the input sequences. Obviously, the tracking is likely to succeed when the sequences are simple. Thus, we have chosen our test sequences according to three difficulties :

- The average number of persons in the scene. This value can range from 1 or 2 for the simplest sequences to more than 10 in very difficult sequences. We want the tracker to be robust until the scene is overcrowded.
- The detection quality. A good tracker must be able to handle the diverse problems encountered by the detection algorithm like shadows, reflections, low target contrast, etc.
- The number of crossings between persons. It is mandatory to have frequent and long crossings in the video in order to assess the reliability of the tracker since single person tracking does not represent a great challenge. The duration of the crossings depends on the camera orientation.

Currently, we have selected indoor scenes video sequences from three different applications: a bank agency, a metro platform and an office. Results will be presented for 3 video sequences:

- V1 (200 frames), good detection quality, which contains 2-3 persons and few crossings.
- V2 (580 frames), average detection quality, which contains 4-6 persons and several crossings.
- V3 (500 frames), bad detection quality, which contains 7-11 persons frequently crossing each other.

2.3 Ground Truth Generation

Once video sequences have been selected, we have defined ground truth using a software called ViPER [8].

The ground truth definition is subjective. There are several key questions a user has to answer before he/she can define ground truth. Do we draw the bounding box of the whole person when the person is occluded? Do we draw two bounding boxes when two people walk very close to one another or do we draw only one for the group? These choices have to be made with respect to the target application and the features to evaluate. In counterpart, care must be taken to avoid introducing a bias when defining ground truth if, for example, the assessor is aware of typical tracking errors.

We have chosen to draw the full bounding box for each person even when he/she is dynamically or statically occluded. In this way, we are able to determine whether the segmentation process has correctly labelled the person as OCCLUDED. Since the coordinates of invisible parts are guessed, this choice can lead to imprecisions. But this

is not crucial as our target application does not require a precise recovery of the shape.

There are two different types of ground truth: one for the segmentation (explained above) and one for the F2F tracker. The latter consists in giving the same identifier to a person throughout the whole sequence. Thanks to these two ground truths, we can quantify the degradation of the tracker results induced by bad detection. We simply have to compare the output of the tracker when we feed it with the output of the detection or with the segmentation ground truth. It is also a way to check how good the tracker is at solving detection problems.

2.4 Global Evaluation Algorithm

Evaluation is done by using a supervised technique which compares tracking outputs and ground truth. The goal of this global evaluation is to rank tracking processes by giving a global analysis of their performances. The proposed method can be applied to most trackers for two main reasons. First, both inputs are required to be coded in standard XML. This is a widespread and easy-to-use format. Second, this method only uses 2D information to produce the classification of errors. Of course, the more featured the evaluator inputs are, the more precise the classification is.

The evaluation algorithm consists of two parts: motion detection evaluation and F2F tracking evaluation. The algorithm classifies detections made by the motion detector and links made by the tracker into three main categories: true positives (TP), false positives (FP) and false negatives (FN). True negatives (TN) are of no interest here. Each main category has several sub-categories nuancing the results. In order to facilitate the diagnosis, all results are further categorized according to the type of occlusion (static or dynamic). When there is no occlusion, results are broken down according to the camera distance (close, medium, far). Static occlusion happens when people, represented by their bounding boxes, are occluded by the static inventory of the scene (e.g., walls, furniture, etc), while dynamic occlusion occurs when people overlap. The two types of evaluation are described separately in the following.

2.4.1 Motion detection evaluation

The classification into positives or negatives depends solely on the degree of overlap between the ground truth objects and the bounding boxes made by the system. A false negative is a ground truth object not sufficiently covered by a detected bounding box. A false positive is a detected bounding box not covered or not sufficiently covered by any ground truth object. False positives are registered in the same way as the false negatives.

The detections which are not false negatives or false positives are true positives, i.e. bounding boxes sufficiently overlapping a ground truth object. For a true positive detection we register whether the 2D form of the bounding box agrees with the corresponding ground truth object, and also whether the 3D centres of gravity conform well. In addition we register to what degree the system chooses the right class (PERSON or GROUP_OF_PEOPLE) for the box. Finally we register whether the system correctly detects the presence of static occlusion (OCCLUDED_PERSON).

video V2	TP (class)	FN	FP
static occl.	720 (305)	14	0
dyn. occl.	393 (49)	0	0
close	7 (7)	0	0
medium	699 (562)	7	24
far	0 (0)	0	9
total	1819 (923)	21	33

Table 1. Motion detection: true positives (TP), false negatives (FN) and false positives (FP) for video sequence V2

Results are illustrated in table 1 for video V2. The first column show the number of true positives. The number in parentheses indicates the number of true positives with a good class label. The two last columns show the number of false negatives and false positives. Rows correspond to the classification by occlusion type or camera distance. The purpose of motion detection evaluation is to verify that the chosen video sequences are sufficiently varied, and that they pose problems of differing nature for the detector and the tracker (e.g., the class is wrong under occlusion). These evaluation results also serve to provide an impression of the overall performance of the segmentation procedure, and thereby a notion of the difficulties faced by the subsequent F2F tracking.

2.4.2 F2F tracker evaluation

For the F2F tracker, the classification into positives or negatives depends both on the degree of overlap between the ground truth objects and the bounding boxes made by the system, and also naturally on the presence of links between the boxes.

A true positive link is a link created by the system combining two bounding boxes that both sufficiently cover a ground truth object at times t and $t+1$. For a true positive link, we register to what degree the two boxes represent a good detection of the underlying ground truth object. Imprecise detection is essentially a segmentation problem rather than a tracking problem. Though, it is studied in order to assess the frequency of occurrence in the links built by the system. We also register whether the link made is the tracker's first choice (i.e. highest valued) of all the links associated with the two bounding boxes. A second link occurs when the tracker computes several links and its second choice corresponds to the ground truth link. True positive link evaluation is not very useful in terms of identifying tracker errors, but gives an interesting overall view of the tracker's confidence in its choice of links. We illustrate the

TP	Partial detection	Good detection	2nd link	1st link
static occl.	0	396	13	383
dyn. occl.	1	359	28	332
close	0	46	0	46
medium	1	783	11	773
far	0	0	0	0
total	2	1584	52	1534

Table 2. F2F tracking: true positives (TP) for video sequence V3

	V1: few people		V3: many people	
FN	Partial det.	Good det.	Partial det.	Good det.
static occl.	0	3	50	114
dyn. occl.	14	0	70	9
close	0	0	8	0
medium	3	0	91	21
far	0	0	0	2
total	17	3	219	146

Table 3. F2F tracking: false negatives (FN) comparison for video sequences V1 and V3

true positive for a rather difficult sequence (V3) in table 2. The two first columns show the number of links which are made between bounding boxes that are partially or well detected, respectively. The two last columns show the number of links which are the tracker's second or first choice, respectively. We can observe that most of the links which have been found are first links. Obviously, the tracker has much more difficulties in the presence of occlusions.

All links made by the system that are not true positive are classified as false positive. For a false positive link between two bounding boxes, we register whether the boxes correspond to people or noise. This gives three subcategories of false positive links: person-person, person-noise and noise-noise (where the objects are different).

A false negative link is a link missed by the tracker. This is due to either partially detected bounding boxes (at time t or $t+1$ or both) or a missing link between correctly detected bounding boxes. For a false negative link, we register whether the corresponding ground truth object is well detected or not.

In table 3, we show a comparison between video sequences V1 and V3, for the number of false negatives. Degradation of performance is clearly visible for the most difficult sequence.

The chosen categories reflect interesting characteristics of the link and facilitates subsequent identification of the tracker shortcomings. Most interesting in terms of tracker improvement are the false negative links. For each identified error, the system produces a text file containing the frames where this error is present.

	Motion Detector			F2F Tracker		
	TP	FN	FP	TP	FN	FP
V1	98.5%	1.5%	7%	90%	10%	1%
V2	98.8%	1.2%	1.8%	84%	16%	4.4%
V3	94%	6%	8%	81.2%	18.8%	5%

Table 4. Motion detection and F2F tracking: true positives (TP), false negatives (FN) and false positives (FP) comparison for video sequences V1, V2 and V3

To conclude this section, we illustrate in table 4 the comparison of true positives, false negatives and false positives for the three test sequences. We have represented the detection rate (TP/(TP+FN)) for true positives, the false negative rate (FN/(TP+FN)) for false negatives and the false positive rate (FP/(FP/TP)).

3 Fine Evaluation Algorithm

The goal of the fine evaluation is to improve the tracking process by classifying and grouping the errors of the tracking algorithm. There are two situations where the global evaluation process is not sufficient:

- Different errors of the tracking algorithm are classified as one error type.
- Similar errors are classified into different error types.

In the first case, the global evaluation algorithm has to be modified to discriminate the given error type into more accurate subtypes by refining the existing evaluation criteria. The second case occurs when the classification does not match the real tracking errors. In this case, the error types have to be redefined using new criteria.

In our case, we have focused only on the refinement of the F2F tracking evaluation since motion detection evaluation results are very good. The global evaluation shows different types of errors classified as false negatives. We have refined these errors into four subtypes:

- Split of a person into several body parts. This error is due to an over-segmentation of a person: at time t the person is detected as one moving region and at time $t+1$ as two (or more) moving regions corresponding to different body parts (e.g., head, body, feet). Usually in this situation, the main body part is tracked and the remaining parts are lost.
- Merge of body parts of a person into a well-detected person. This situation is similar to the previous one.
- Split of a group of people into distinct persons. Usually in this situation, one of the person is isolated and detected by a large moving region which is close to the detection of the group at the previous time. Then, the moving regions corresponding to the remaining persons are lost.

	Person		Group	
	Split	Merge	Split	Merge
V1	1	1	7	7
V2	20	12	92	108
V3	25	25	61	82

Table 5. Tracking errors for video sequences V1, V2 and V3

- Merge of separated persons into a group of persons. This situation is similar to the previous one.

Table 5 describes the four tracking errors subtypes. After analyzing all these error situations, we have found out that these four subtypes correspond to specific tracking errors.

4 Evaluation Utilization

The first step to be able to use the evaluation results is to isolate an error type by refining the global evaluation process as defined above.

Once we have a satisfactory classification of the tracker errors, we are able to fix the tracking algorithm. For each type of error, we browse the different instances of this error and we try to determine if the problem can be solved by an adequate change in a tracking parameter or if we need to introduce additive knowledge.

Finally, the last step consists in re-evaluating the tracking process, first using the same set of video sequences and ground truth and second by extending this set with more challenging videos.

4.1 Repair of the Tracker

In the previous sections, we have shown how we isolated tracking errors. In this section, we show how we have repaired our tracking algorithm by analysing when and how the merge and split situations occur. These situations occur when a subset of N called N_s is in relation with a subset of O called O_s. These subsets of neighbour moving regions are called clusters of moving regions (CMR). Two moving regions of N are neighbours if they are both close to a moving region of O using a coarse 2D distance metric. Moreover, two moving regions are neighbours if they share a common neighbour with a third moving region (transitivity rule).

4.1.1 Discrimination between concurrent hypotheses

We generate different types of hypotheses to estimate which situations can appear within a CMR couple. A hypothesis corresponds to a phenomenon of the real world and attempts to explain an association between zero, one or several moving regions of the CMR detected at time t, and zero, one or several moving regions of the CMR detected at time $t+1$. For this reason, hypotheses are only computed for the moving regions classified as PERSON or GROUP_OF_PEOPLE. There are five types of hypothesis: *enter*, *exit*, *continue*, *split* and *merge* which are computed as follows. For each moving region in a CMR, we register the number of moving regions in the other CMR whose bounding boxes are at least α percent in overlap, α being a parameter of the algorithm. This number determines the hypothesis type. For instance, for an old moving region we will compute an *exit* hypothesis if this number is 0, a *continue* hypothesis if this number is 1 and a *split* hypothesis if it is more than 1.

Once possible hypotheses within the CMR couple have been determined, we evaluate accurately each hypothesis using additive knowledge such as contextual knowledge and 3D information. We have defined a function f: $O \times N \rightarrow \mathbf{R}$ between two moving regions which compares their 3D size, their type, their 2D distance and their 3D distance. This is the sensitive function of the algorithm and it has 11 parameters. This function is used to compute the likelihood $L(h, f)$ for each hypothesis h. The output range for L is $[0..100]$.

Finally, we choose the best scored hypothesis and we remove the processed (linked) moving regions from the CMR couple. We iterate this procedure until the CMR couple is emptied. We show hereunder some cues about the computation of the likelihood of each hypothesis type.

4.1.2 *enter* and *exit* hypothesis

Two situations can appear in an *exit* hypothesis: 1) the moving region o_a is on the camera border or on a contextual in/out zone, 2) the moving region is not exiting. In the former case, the score will be very good (i.e. $L(\textit{exit}, f) = 100$) as no other available information (e.g., temporal one) at that processing stage can lead us to another conclusion. In the latter case, the score will be bad unless the moving region matches a noise n_b in the new frame. In this situation, we have $L(\textit{exit}, f) = f(o_a, n_b)$. The *enter* hypothesis is similarly solved.

4.1.3 *continue* hypothesis

In a *continue* hypothesis, we compute the score between the old moving region o_a and the new moving region n_b: $L(\textit{continue}, f) = f(o_a, n_b)$. We then try to further improve the score if there are some moving regions classified as NOISE in the CMR of frame $t+1$. We test whether the likelihood improves when we merge a noise with the person of the same frame. The better the merge is (if any) the better the hypothesis score is.

4.1.4 *split* and *merge* hypothesis

Three main situations can occur for the *split* hypothesis: 1) we have a group at time t that splits up at time $t+1$, 2) we have a person a time t who is segmented in several parts at time $t+1$, 3) we have a group at time t that splits up at time $t+1$ with a person segmented in several parts. In addition, we can have the three previous situations with either a noise classified as a person.

Since we have one old moving region o_a, we first determine the best corresponding new moving region n_j such that $L(split, f) = max_j f(o_a, n_j)$. Then, as long as the score improves, we try to merge iteratively n_j with one of the remaining new moving regions: $L^1(split, f) = max_k f(o_a, n_j \cup n_k)$. If $L^1 > L$ then $n_j = n_j \cup n_k$ and we iterate. During this process, we register all the involved new moving regions.

Second, a 3D distance criterion helps us know which of the registered new moving regions comes from an over segmentation of a person. The basic idea is that the 3D distance between several parts of the same person will be high as there will be a bad projection. So we are able to distinguish which are the moving regions to be merged and which are the moving regions splitting up from a group. Noise is hardly detectable.

The *merge* hypothesis is similarly solved.

4.2 Discussion

Using the fine evaluation process we have isolated four types of errors leading to a systematic bias of the algorithm. For instance, when a person splits up into several body parts, a moving region is often lost. Thanks to the evaluation process, we refined the tracking algorithm to be able to adjust the parameters of the function f.

We have specified a set of parameters for the computation of each hypothesis such as the *split* hypothesis. All the hypotheses are now processed separately, thus enabling a correct processing of split and merge situations.

The studied tracker is now able to track correctly multiple people in cluttered environment as can be seen on the web page http://www-sop.inria.fr/orion/personnel/Benoit.Georis/index.html.

During this process, it is of prime importance to solve the general problem without depending on one sequence. It is only possible to do so if the video sequences used for the evaluation are diverse and significant enough.

Moreover, this technique could be used in the particular case of a new camera setup. The tracking algorithm could be adjusted and parameterized according to ground truth until acceptable performances are reached. After this configuration mode, the tracker would run without supervised evaluation.

We think that this approach can evolve towards an automatic repair technique of the tracking algorithm. Nevertheless, it requires a deep understanding of the tracking algorithm (e.g., how to parameterize it).

5 Conclusion and Future Work

We have presented a general framework for the evaluation, the diagnosis and the improvement of tracking systems. Tracking algorithms usually contain many parameters and have to be validated on a large number of video sequences. Tuning manually these parameters to optimize tracking algorithms is difficult. We proposed a methodology to improve tracking performances using an evaluation and diagnosis process. This method has currently been applied to human tracking in a video surveillance application context. It has been shown that this approach is able to solve several tracking errors. At this time, we are extending the method to other parts of the system.

The next major goal to achieve is the design of an automatic repair process. For example, future work will emphasis on automatic parameters tuning or automatic subcode selection.

Acknowledgements: This collaboration between INRIA and UCL has been possible and has been granted by the Walloon Region under the FIRST EUROPE program.

Many thanks to Tony Krakenes who designed the evaluation algorithm.

References

[1] C. Wren, A. Azarbayejani, T. Darell, and A. Pentland, "Pfinder: Real-time tracking of the human body," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 780–785, 1997.

[2] A. Baumberg and D. Hogg, "An efficient method for contour tracking using active shape models," in *Proc. of IEEE Workshop on Motion of Non-Rigid and Articulated Objects*, (Austin), pp. 194–199, 1994.

[3] I. Cox and S. Higorani, "An efficient implementation of reid's mht algorithm and its evaluation for the purpose of visual tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, 1996.

[4] T. Ellis, "Performance metrics and methods for tracking in surveillance," in *Proceedings of the Third IEEE Workshop on PETS*, (Copenhagen), June 2002.

[5] J. Ferryman, ed., *Proceedings of the Second IEEE Workshop on PETS (PETS'2001)*, (Kauai, Hawaii), December 9 2001.

[6] F. Cupillard, F. Brémond, and M. Thonnat, "Behaviour recognition for individuals, groups of people and crowd," in *IEE Proc. of the IDSS Symposium - Intelligent Distributed Surveillance Systems*, (London), February 2003.

[7] I. Haritaoglu, D. Harwood, and L. Davis, "W4: Real-time surveillance of people and their activities," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 809–830, August 2000.

[8] http://lamp.cfar.umd.edu/media/research/viper/.