

EFFECTS OF PARAMETERS VARIATIONS IN PARTICLE FILTER TRACKING

Xavier Desurmont^a, Caroline Machy^a, Céline Mancas-Thillou^b, Derek Severin^a, Jean-François Delaigle^a

^a Multitel A.S.B.L., Parc Initialis, Av Copernic, 1, B-7000, Mons, Belgium.

^b Faculté Polytechniques de Mons, Mons, TCTS, Av Copernic, 1, 7000 Mons, Belgium.

ABSTRACT

Many implementations of visual tracking have been proposed since many years. The lack of standard evaluation process has prevented fair comparison between them. In this paper, we simply propose to evaluate different particle filter methods in people tracking applications. We introduce an objective metric and give results according to different parameter variations. Finally, based on our evaluations, we can propose a new particle filter configuration that outperforms other current implementations.

1. INTRODUCTION

The number of applications using video monitoring for security, safety, traffic analysis, marketing and also semantic content retrieval, is increasing, [1] [2]. One basic sub-feature of these systems is to track objects within the visual scene. Different approaches exist to perform this function [3]. In this paper, we aim to evaluate solutions based on particle filtering.

The paper is organized as follows: Section 2 introduces the top-down tracking and specifically the particle filter tracking. Section 3 presents the performance evaluation approach and gives the results comparing the algorithms. Finally, section 4 concludes and indicates future work.

2. TOP DOWN TRACKING

2.1. Top-down / Bottom-up tracking

Top-down tracking consists in estimating the position and characteristics of an object of interest (the target) in the current video frame, given its position and characteristics in the previous frame. It is thus, by definition, a recursive method.

On the contrary, the bottom-up approach consists in segmenting in each frame the moving objects and trying to match them over time. The most restrictive hypothesis of the bottom-up approach is the fixed background requirement. Although moving background estimation is possible, it is usually very time consuming and leads to noisy results.

In the top-down approach, hypotheses of object presence are generated and verified using the data from the current

image. In practice, a model of the object is available and is fitted to the image. The parameters of the transformation are used to determine the current position of the object.

Some techniques representative of the top-down approach for tracking are the Lucas-Kanade algorithm [4], the Mean-Shift algorithm [5] and the particle filter tracking, they are explained in [6]. They differ in the type of features extracted from the frame as well as in the tracking technique (deterministic or stochastic). In all cases, a model of the target is assumed to be available. A human operator can provide the model manually: the kind of application is "click and track". The operator chooses a person or object to track in the image. The selected region is then used to compute the target model. Alternatively, it can come from a detection step [7]: once an object is detected by bottom-up techniques described in the above section, a model or a template of the target can be extracted automatically from that blob.

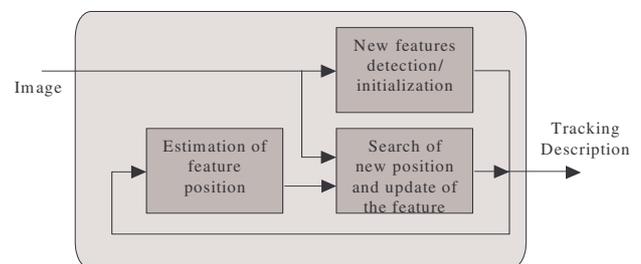


Fig 1. The architecture of the top-down approach.

2.2. Particle Filter Tracking

Particle filters (PF) are sophisticated model estimation techniques. The PF aims to estimate a sequence of hidden parameters x_t based only on the observed data z_t . In the case of object tracking, the state of the tracked object is described by the vector x_t while the vector z_t denotes all the observations $\{z_1, \dots, z_t\}$ up to time t . The idea is to approximate the probability distribution by a weighted sample set: $S = \{(s^{(n)}, \pi^{(n)}) \mid n=1 \dots N\}$. Each sample s represents one hypothetical state of the object, with a corresponding discrete sampling probability π .

The evolution of the sample set is described by propagating each sample according to a system model. Each element of the set is then weighted and N samples are created, by choosing a particular sample with probability $\pi^{(n)} = p(z_t \mid X_t = s_t^{(n)})$.

So particle tracking can be resumed in five main steps:

- 1) **Initialization:** Creation of the set of initial samples and definition of the system equation:

$$x_t = Ax_{t-1} + w_{t-1} \quad (1)$$

- 2) **Prediction:** With the particles from the previous frame $x_{t-1}^{(i)}$ and (1), predict the next positions of the particles.

- 3) **Update weights:** Compute the weight of each particle:

$$\pi_t^{(i)} \sim \exp\left(-\frac{(1-\rho^{(i)})^2}{2\sigma^2}\right) \quad \text{with} \quad \sum_{n=1}^N \pi^{(n)} = 1 \quad (2),(3)$$

- 4) **State Estimation:** The state estimation of an object is estimated at each time step by:

$$E[x_t] = \sum_i \pi^{(i)} x_t^{(i)} \quad (4)$$

- 5) **Resampling:** Particles with high weight generate many particles and particles with low weight are killed.

2. OBJECT MODEL

Many different target model types have been proposed. Rigid templates are useful if large deformations are not expected. Color histograms may be more appropriate in the case of non-rigid motion. However, color has also serious drawbacks because it varies drastically with illumination and is poor in information when people wear dark clothes. The appearance of objects in images depends on many factors such as illumination conditions, shadows, poses, contrast effects, etc. This is why it is often necessary to use adaptive models, i.e. models of targets that can evolve in order to adapt to appearance changes. On the other hand, it is also interesting to use a model independent of these changes.

In this paper, an object is composed of different features (like head, chest, knees...). Consequently, we have the possibility to attribute either 1 or 5 tracks per object, each one representing an interesting feature. Each track contains a PF with its own set of samples. At each iteration, a test is made to check if one or more of the tracks are lost. If yes, a new track is created at the barycenter of the other ones. We used 5 tracks, because it allows a vote that has always a majority (ex: 3,4 or 5). This method is used to be more robust to occlusions and appearance changes.

3. PERFORMANCE EVALUATION

The goal of performance evaluation is to determine whether a system answers correctly to some problems (defined as a set of inputs and outputs). Please refer to [11] for a review on general performance evaluation of vision systems. We first introduce the parameters of the PF. Then, we propose a metric to give a score to each configuration of parameters.

Next, we present the video dataset we are using. Finally, we report the results and try to explain them.

Implementations of the particle filtering have already been evaluated: two basic ones with color histogram in [8],[9] and the same one added with a selection of few improvements in [10]. But, except for [10], the comparison did compute the enhancement provided by the addition of new algorithm features.

3.1. Algorithms and parameters

Our implementation of the object tracking is based on the Bayesian Filtering Library [12] and implements the method described in [13]. Our algorithm is organised as follows:

- 1) **Initialization of the tracks** (position and reference feature) with Harris filter [14] which gives one interesting feature per track around the initialization point. Creation of the sets of particles in sampling the positions with the prior noise.
- 2) **Update weights:** The Battacharayya distance [15] is used to compute the distance between the histogram of the current feature and the reference feature, and then the equation (2) gives the weights.
- 3) **State estimation:** An estimator of the mean (4) is used for the estimated position.
- 4) **Resampling**
- 5) **Update reference feature**

Parameter variations:

Number of tracks: 1 or 5 tracks per target.

Colors: two implementations of color spaces have been realized: luminance (no color) and RGB. The distance function is thus different.

$$L = \text{luminance} = 0.3 R + 0.59 G + 0.11 B \quad (5)$$

Appearance model: distributions (histograms), as well as blocks of pixels are tested as target models.

Motion model: the model of motion is a bootstrap filter. This is a particular implementation of a PF in which the proposal density is equal to the pdf describing the system model. The model can be expressed as: $x_t = Ax_{t-1} + w_{t-1}$, A describes the evolution of the system and w the noise of the system.

The size of the system is also a parameter: with value 2 (the system is only described with the position of the particles) or 4 (the system is described with position and speed of the particles).

Search: We propose a feature that enables the particles to do a search around their position to find the best match in the image. The PF is then able to perform a deterministic search like in Lucas-Kanade [4] or Mean-Shift algorithm [5].

Then, we also have the possibility to update the position of the particle to the best match position.

Resampling: We finally propose to force the resampling at each iteration. Indeed, in BFL, the resampling is made only if necessary.

3.2. Metric

We must determine an objective criteria to evaluate the good functionality of the system that will be derivated in metrics. A metric is an algorithm that receives the ground truth (GT) and the results (RES) from the system. The output of the metric is the information of the difference between GT and RES. It can be qualitative (e.g. good detection) or quantitative (e.g. mean size error). The GT and the RES are expressed with meta-data. In our case, we have a set of positions and identifiers of people.

As a metric, we report the percentage of time until the tracking system loses the target (time of tracking / time of presence in the GT). We define the loss of the target when the Euclidian distance between the GT and the RES of a target in the 2D frame is above a defined threshold. For the experiments, we choose 30% of the height of the image. Note that the initialisation of the tracking is done by use of the GT as we are not considering the detection process. The metric is explained with the figure below:

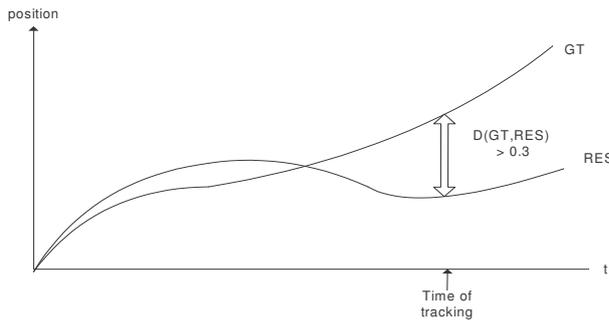


Fig 2. When RES diverges from GT, the distance is increasing.

3.3. Video datasets

Some videos are shot with fixed cameras and others are from Pan-Tilt-Zoom (PTZ) cameras. We are using the Caviar [16] test sequences as well as TricTrac [17] ones. They are in color at 25 fps.



Caviar1 sequences



Trictrac sequences

Fig 3. Video datasets used in the evaluation.

3.4. Results

Tab 1. Result of parameters variation.

Duration of GT track	Number of tested tracks	parameter = imageformat		Improvement %
		luminance	rgb	
0-5s	1408	0,74	0,76	2,76
5-10s	734	0,56	0,69	23,98
10-20s	1119	0,44	0,45	2,47
20+s	337	0,12	0,22	83,12

Duration of GT track	Number of tested tracks	parameter = feature		Improvement %
		block	histogram	
0-5s	0	0,00	0,00	N/A
5-10s	192	0,45	0,46	2,20
10-20s	384	0,18	0,33	82,49
20+s	192	0,21	0,14	-31,44

Duration of GT track	Number of tested tracks	parameter = resampling		Improvement %
		False	True	
0-5s	1242	0,74	0,76	2,92
5-10s	987	0,59	0,60	3,09
10-20s	1399	0,37	0,37	0,71
20+s	523	0,20	0,22	7,94

Duration of GT track	Number of tested tracks	parameter = nboftrackspertarget		Improvement %
		1	5	
0-5s	1720	0,73	0,80	10,04
5-10s	1147	0,57	0,63	10,82
10-20s	1642	0,35	0,43	24,35
20+s	560	0,18	0,24	32,45

Duration of GT track	Number of tested tracks	parameter = statesize		Improvement %
		2	4	
0-5s	1432	0,76	0,75	-2,01
5-10s	1024	0,61	0,58	-4,53
10-20s	1445	0,37	0,39	6,89
20+s	549	0,21	0,21	-1,27

Duration of GT track	Number of tested tracks	parameter = usesearch		Improvement %
		False	True	
0-5s	903	0,74	0,78	5,49
5-10s	680	0,53	0,61	15,30
10-20s	923	0,34	0,42	24,43
20+s	363	0,18	0,24	32,92

Duration of GT track	Number of tested tracks	parameter = updateposition		Improvement %
		False	True	
0-5s	907	0,74	0,75	2,18
5-10s	680	0,53	0,65	22,63
10-20s	923	0,34	0,39	15,50
20+s	363	0,18	0,21	17,57

Results are given for each parameter variation. They are split between different GT durations. Indeed, a short duration displacement in the scene is usually well tracked because the system does not have time to loose the target. In order to show the improvement of a parameter change, it is better to use longer durations (like 10 seconds and over). In

the presented tables, the parameter is given (e.g. imageformat) as well as the different values of it (e.g. luminance and rgb). Then the results are the means relative duration of the RS tracks comparing the GT tracks, which gives a percentage of tracking. The improvement is the comparison of the two values.

3.5. Interpretation of the results

Many results are going on the expected way. The handling of color enhances the tracking, indeed colors allow to discriminate some features that look the same in luminance. The use of multiple targets for a single object improves also the result, because it handles theoretically 40% of losses (votes of the 5 tracks; in case of disagreements, it cancels the 1 or 2 worse tracks).

The "search" capability also increases the duration of good tracking as well as the update of the position. In fact, this should be compared to the result of an increase of the number of particles. The result may be the same.

For the other parameters, it is more difficult to bring out conclusion. The re-sampling at each frame also improves slightly the quality of the tracking. This is due to a better fit because of the higher precision of the pdf.

The size of the motion model gives worse results when we consider the speed added to the position. This is maybe because a human is not as predictable as a car in terms of motion.

For the feature (histogram or block of pixel), it is also difficult to find out which one is the best. Both have advantages and drawbacks, histogram is more robust (the pdf is usually smoother), but block is more accurate in the position. So block is maybe better when there is less noise in the sequence. Block may also be worse when sampling of particle is lower (possibility to degrade the finding of the local minimum).

4. CONCLUSION AND FUTURE WORK

We made a wide evaluation of different configurations of PF people trackings. Thus we can propose our best solution: 5 features per object, rgb histogram, with two values displacement model and a search-and-update function.

Despite the fact that we evaluated on different video sequences, these results are specific to a given application and thus the choice of a given PF for one particular method depends on the requirements of the final application.

In future works, we will evaluate also other features (co-occurrence matrix and gabor descriptors), resource consumption of each parameter change and more tracking algorithms like bottom-up and also on other types of contexts like crowded scenes.

Acknowledgment: This work is supported by the Walloon Region within the scope of the ITEA CANDELA project.

5. REFERENCES

- [1] A. Cavallaro et al, "Segmenting moving objects : the MODEST video object kernel", Proceedings of Workshop on Image Analysis For Multimedia Interactive Services, May 2001.
- [2] F. Cupillard et al, "Tracking groups of people for video surveillance", Proc. of the 2nd European Workshop on Advanced Video-Based Surveillance Systems, London, September 2001.
- [3] I. Lutkebohle, "Capabilities and Limitations of Visual Surveillance", 22nd Chaos Communication Congress, Berlin, Germany, December 27th to 30th, 2005.
- [4] B. D. Lucas and T. Kanade, "An Iterative Image Registration Technique with an Application to Stereo Vision", International Joint Conference on Artificial Intelligence, pp. 674-679, 1981.
- [5] D. Comaniciu, V. Ramesh, P. Meer, "Real-Time Tracking of Non-Rigid Objects using Mean Shift", IEEE Conf. Computer Vision and Pattern Recognition, Vol. 2, pp.142-149, 2000.
- [6] X. Desurmont et al, "Chapter 5: A General-Purpose System for Distributed Surveillance and Communication", Intelligent Distributed Video Surveillance Systems (S.A Velastin & P Remagnino Eds.), IEE, London, ISBN: 0-86341-504-0, 2005.
- [7] I. Martínez-Ponte et al, "Robust human face hiding ensuring privacy", 6th International Workshop on Image Analysis for Multimedia Interactive Services, Montreux, Switzerland, 2005.
- [8] K. Nummiaro, E. Koller-Meier, L. Van Gool, "Color features for tracking non-rigid objects", Special Issue on Visual Surveillance, ACTA Automatica Sinica, 2003.
- [9] J. Czyz et al, "A color-based particle filter for joint detection and tracking of multiple objects", IEEE International Conference on Acoustics, Speech and Signal Processing, 2005.
- [10] A. Jacquot, P. Sturn, O. Ruch, "Adaptive tracking of non rigid objects based color histograms and automatic parameter selection", Image and vision computing, pp 99-110, 2005.
- [11] X. Desurmont et al, "Performance evaluation of real-time video content analysis systems in the CANDELA project", conference on Real-Time Imaging IX, IS&T/SPIE Symposium on Electronic Imaging 2005, San Jose, CA USA, January 2005.
- [12] BFL : <http://people.mech.kuleuven.be/~kgadeyne/bfl.html>
- [13] K. Nummiaro, E. Koller-Meier, L. Van Gool, "An adaptive color-based particle filter", Symposium for Pattern Recognition of the DAGM, Zuerich, September 2002.
- [14] C. Harris, M. Stephens, "A combined corner and edge detector", Proceedings of The Fourth Alvey Vision Conference, Manchester, pp 147-151. 1988.
- [15] S.H. Cha, S.N. Srihari, "On measuring the distance between histograms", Pattern Recognition, Vol. 35, No. 6., pp. 1355-1370, June 2002.
- [16] The data as coming from the EC Funded CAVIAR project / IST 2001 37540, www.dai.ed.ac.uk/homes/rbf/CAVIAR/, 2005.
- [17] X. Desurmont, J-B. Hayet, J-F. Delaigle, J. Piater and B. Macq, TRICTRAC Video Dataset: Public HDTV Synthetic Soccer Video Sequences With Ground Truth. Workshop on Computer Vision Based Analysis in Sport Environments (CVBASE), 2006.