A Review on Particle Filter for a Video Based Tracking Problem

Report Author: Alfredo Reichlin
Group Partner: Luca Marson

Abstract—Tracking of object moving along highly non linear trajectory is a challenging task and the Particle Filter is a valid choice for solving this problem because it is able to deal efficiently with multimodal distributions and non linear motion models. This project summarizes different ways of integrating color feature and edge detections to implement a robust estimator for the tracking problem. Color distribution are robust to partial occlusions and are rotational and scale invariant, but they perform poorly when the background is similar to the object to be tracked. Edge detection on the other hand is robust to this kind of problem, but it can be easily tricked by shapes similar to the reference one, like the object shadow

1 Introduction

Video tracking has different interesting applications like video surveillance, video compression, virtual reality. Particle filter has several important advantages over other estimation method and it's ideal for video tracking due to its high nonlinearity, adaptivity and multihypotheses.
Implementing the Particle Filter to solve the tracking problem means finding a suitable motion model, as well as different osservation models. We followed two different approach: one which consists in considering a cloud of particle representing single pixel on the video frame, and another which assing a subregion of the frame to each particle. We experimented with both color and edge detection as suggested in [3].

1.1 Outline

In section 2 a brief summary of related works is presented. In section 3 the particle filter algorithm is presented from both a theoretical viewpoint and our practical implementation. In section 4 Results of the experiments we made are presented along with some considerations. In the Conclusive section some general considerations are made and possible ways to improve the algorithm are stated.

2 Related work

The use of particle filter to solve tracking problems has been already studied in detail. [4] where the first to introduce the condensation algorithm to the tracking problem, which have have been extended in [8]. Color based particle filter have since been introduced, [9, 10], with overall satisfactory results and some serious limitations like a change in lighting and occlusions. Other observation models as edge and texture detection have been proposed.

3 Particle filter

Particle filter is a non parametric implementation of the Bayes filter where the posterior is approximated by a finite number of parameters. Each particle is thus a weighted hypothesis as to what the true world state may be at any time step. Particle filter makes no assumption on the posterior distribution and can use highly non linear motion models making it more robust than a Kalman filter. However it has the disadvantage of approximating the belief by a subset of samples making it subject to the problem of particle deprivation. Moreover the number of particles used influence both the complexity of the algorithm and the accuracy. At each time step each sample is moved according to a prediction model and then weighted according to an observation function.

$$w_t = p(z_t \mid x_t) \tag{1}$$

To avoid having particles in low probability regions a resampling algorithm is used to concentrate the particles on the probable hypotheses. The belief is then computed weighting all N hypotheses:

$$belief = \frac{1}{N}\sum_{i=1}^{N} w_i x_i \tag{2}$$

3.1 Frame Representation

Digital image frames can be represented in different ways. In the RGB representation each pixel is a vector of three numbers indicating the intensity of red, blue and green. In the HSV representation, instead of the three primary colors, the hue, the saturation and the value of each pixel is considered. The latter has the advantage of being more robust to lighting changes. Furthermore in our model we considered also images constructed with the use of a Sobel operator where the intensity of each pixel represent wether or not that pixel is an edge or not.

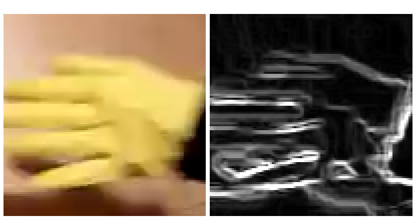

Fig. 1: original and sobel image

3.2 Initialization

We considered both tracking and global localization for the first frame.

3.2.1 *Tracking*

For the tracking problem the initial point is given manually as in [1]. Particles are sampled from a gaussian distribution where the mean is the given starting point and variance. When particles are initialized, the value of the frame in the points where the particles are placed represent the reference used by the observation model. Unfortunately to initialize particles in this way we need to know where the object is, how big it is and it has to be completely visible but it has the advantage of saving the precise values of the object as reference.

3.2.2 *Global Localization*

For the global localization problem particles are sampled uniformly on the entire frame and the reference is manually given. While this method can be used if both the position and the dimension of the object is not known, the reference has to be known a priori.

3.3 Prediction Model

Due to the fact that normally moving objects in video follow a highly non linear trajectory, we made the prediction model as general as possible. As in [2], at every prediction step, particles are scattered around the original belief according to the model:

$$\hat{x}_{t+1} = x_t + r \tag{3}$$

Where r is a random value between the interval $(R, -R)$. To avoid the problem of changing the value of R according to the speed and velocity of the object we update it dynamically by setting it to:

$$R_t = R_{t-1} - R_{t-2} + \eta \tag{4}$$

Where η is a constant used to increase the uncertainty.

3.4 Observation Model

The observation model is probably the most important part for a tracking algorithm. We analyzed different options.

3.4.1 First Observation Model

In this model each particle represent one pixel and the observation is made on the RGB values. The weight of each particle is proportional to the negative exponential of the euclidean distance between the values of the particle and the ones from the reference divided by the observation uncertainty.

3.4.2 Second Observation Model

For the second model, instead of associating only one pixel to each particle, a subregion of the frame is considered as in [1,5]. For each particle, in fact, we build an histogram in 4 dimensions where the basis are the three values of R, G and B and the height is the number of pixels inside the subregion with those RGB values passed through a kernel. The kernel is chosen such that closer pixels have a grater influence. This histogram is then compared to the one used as the reference and the weight is proportional to the negative exponential of the Bhattacharyya distance between the two histograms divided by the observation uncertainty. The Bhattacharyya distance is defined as:

$$\rho[p, q] = \sum_{u=1}^{m} \sqrt{p(u)q(u)}$$

$$d = \sqrt{1 - \rho[p, q]}$$

where m is the number of bins.

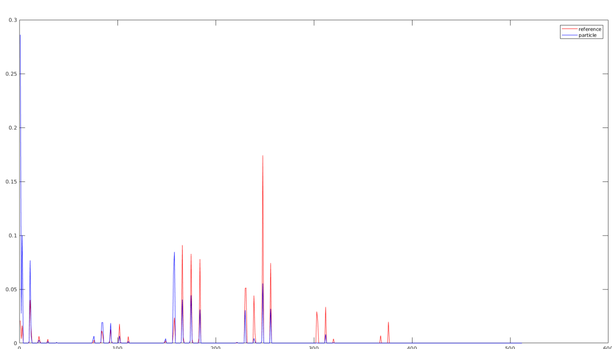

Fig. 2: RGB histogram with linearized base

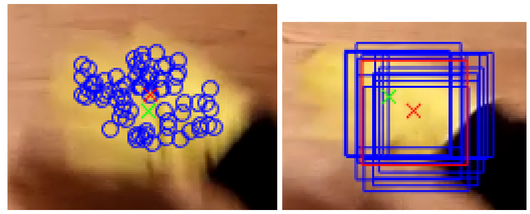

Fig. 3: first model and second model

3.4.3 Third Observation Model

The third observation model we implemented is based on the same principles of the second one with the only difference that instead of using RGB values to build the histogram, the magnitude and arctangent of a Sobel operator is used as in [2]. By applying a Sobel operator to a pixel we get how much and in which direction can that pixel be considered an edge in the image or not. This is a method for tracking edges instead of colors. It has the advantage of being more robust to color change but edges are in general less reliable than color and they can change easily during the video.

3.4.4 Fourth Observation Model

In the fourth model both the first and the third models are joint to make a more robust estimator. In fact, particle's weights are initially computed based on the first model, a random subset of the particles is then chose and the third model is used on them. If some of these particles return a value, based on the edge similarity, that is greater then a certain threshold, all the particles inside those histograms receive a greater weight. By using multiple cues the filter should be able to avoid the weak points of both a color based and an edge based algorithm. This strength is, however, achieved at the expenses of an higher complexity.

3.4.5 Fifth Observation Model

The fifth and final observation model we implemented is based on the same principles of the fourth one with the exception of how the color is represented. In this model we use a HSV representation instead of the RGB. Using HSV should make the filter rely less on the intensity of the color so that a change in light would not be disruptive.

3.5 Resampling

For the resampling step we used a normal multinomial resampling and a low variance systematic resampling described in [7].

3.6 Mutation Rate

To make the estimator robust to total occlusion we introduced a mutation rate. During the prediction step a small number of particles are resampled randomly on the whole frame. In this way if they are in a region with small probability they will be discarded during resample but if instead reach an high probability region, they will make more particles converge there. Using this algorithm, if the object disappears for some time steps and reappear somewhere else, there is a chance to make the filter converge again. The probability of finding the object is:

$$p = 1 - (\frac{A_f - A_{obj}}{A_f})^{\nu n}$$

Where A_f is the area of the frame, A_{obj} is the area of the object, ν is the number of mutated particles at each time step and n is the number of elapsed time steps.

4 Experimental results

Tracking using a particle filter can be subject to different kinds of error depending on what parameters have been used and what models have been chosen. For example when using a color cue based particle filter, the color of the background can have devastating effects on the convergence of the filter. For our experiments we recorded a video with a resolution of 640x480 pixels per frame and 30 frames per second. The object to track is a yellow hand moving around over a light brown background. During the video both a complete occlusion event and a sudden change in the illumination happen in order to maximize the stress on the filter. The problems associated to this video are:

1) background color: In this case even if yellow and light brown are not so different colors, the background does not poses serious threats to the convergence of the filter.

2) Edge recognition: For a substantial part of the video, the hand is well illuminated and a shadow on the floor can be spotted. Having a similar shape to that of the hand so sometimes the filter mistakes the shadow for the hand if the edge cue is used as the observation model

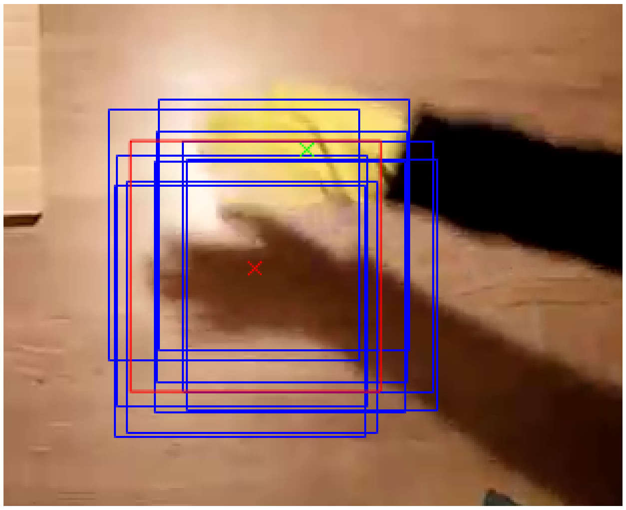

Fig. 4: shadow believed to be the hand

3) Total occlusion: For some frames the hand disappears from the video when it passes behind an obstacle. While the object is not present in the video the filter cannot follow it but it should be able to track it down as soon as it reappears. In this case the mutation rate plays a fundamental role giving the filter the possibility of finding the object.

4) Light change: During the last part of the video, a suddenly the illumination is changed and the color of the object, as well as the one of the background, changes. This event can be problematic to both a color based and an edge based particle filter. In fact for the color one, the difference between the color of the particles and the ones of the reference increases and in same case it can be a problem. Also for the edge based filter it can be a problem, a poor light makes the edges less vivid.

The effects of this four problems can be seen in the figures 7, 8, 9, 10.

Fig. 5: before, during and after the occlusion

Fig. 6: before, during and after the light change

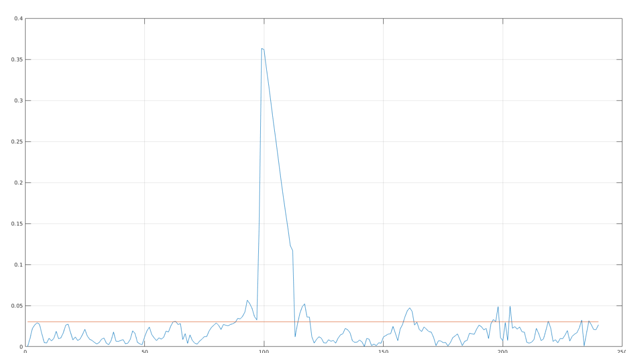

Fig. 7

The blue line represents the euclidean distance between the belief and the groundtruth normalized by the length of the diagonal of the frame in each frame, the red one represent the mean error. As can be noticed, all the plots have a spike around frame 100, that is when the total occlusion happens. No filter is able to follow the object and the error increases drastically. Figures 7, 8 and 9 represent the behavior of particle filters that are able to find the object after the occlusion thanks to the mutation

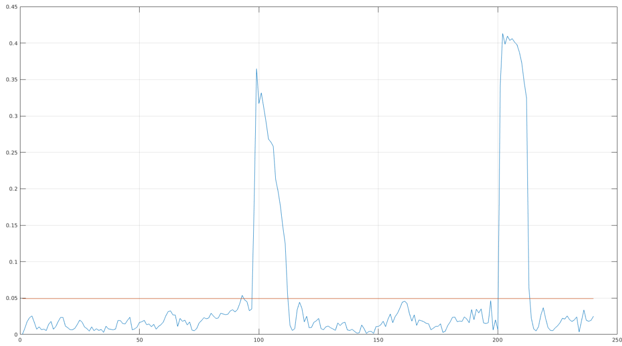

Fig. 8

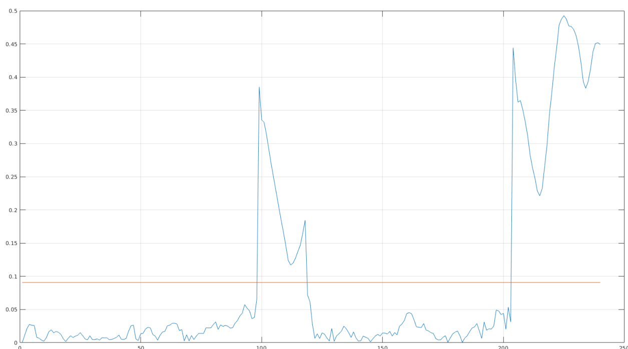

Fig. 9

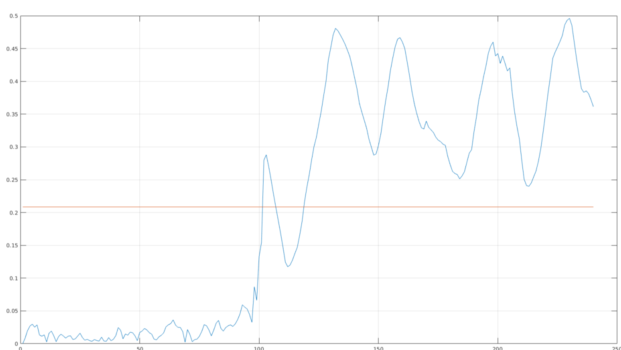

Fig. 10

of some particles while the filter in figure 10 diverges completely. Figures 8 and 9 present another spike around frame 200, that is when light changes.Figure 7, instead, is robust to the change and does not increases the error.

4.1 Models comparison

The different models described in the previous section have been compared using this video.

In figure 11 two filters, one with the first observation model (blue) and one with the second observation model (yellow), are compered. As can be seen the first model performs worse on the occlusion problem but it is more resistant on the changing light. The overall error is lower for the first model.

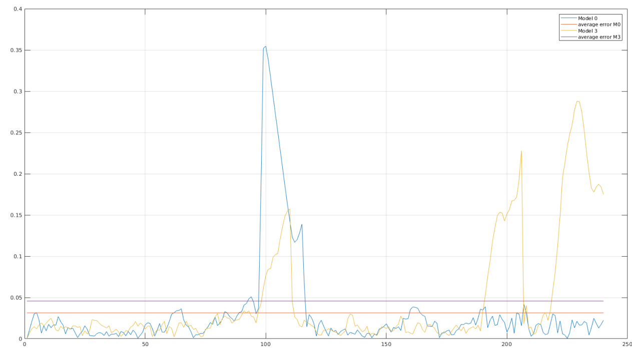

Fig. 11

In figure 12 two filters, one with the fourth observation model (blue) and one with the fifth observation model (yellow), are compered. The performance are almost comparable for the majority of frames. Unexpectedly the fifth model performs worse than the fourth during the change in illumination. However it must be noticed that how the parameters of HSV are tuned can have drastic results on the performances.

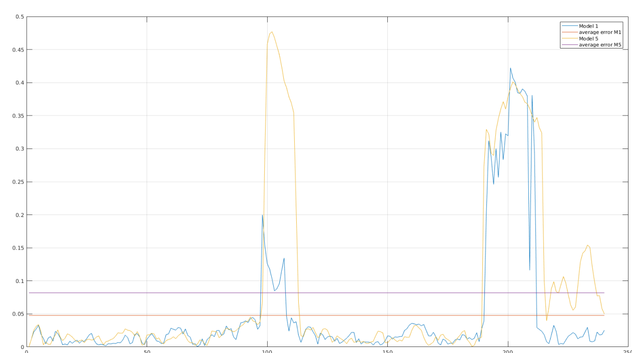

Fig. 12

In figure 13 two filters, one with the second observation model (blue) and one with the third observation model (yellow), are compered. The second model performs undoubtedly better than the third one. The edge based observation model used is, in fact, too simple and edges are often misleading.

In figure 14 filters with a fixed observation model but different mutation rate are com-

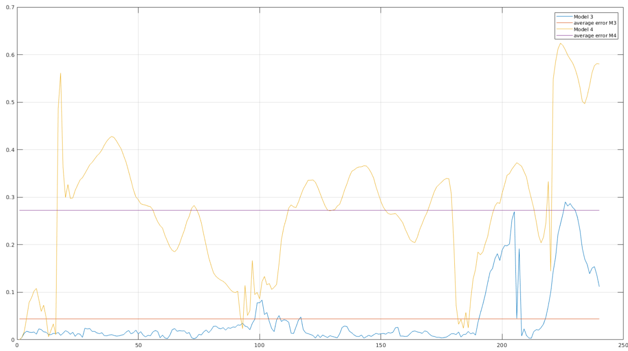

Fig. 13

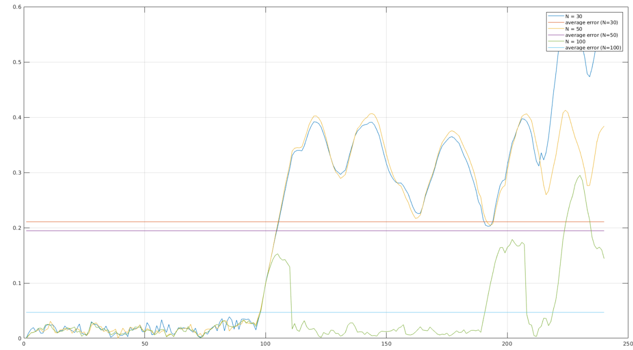

Fig. 15

pared. The blue line has a 0% mutation rate, the yellow one 1% and the green one 10%. As expected the higher the mutation rate the faster the filter is able to relocate the object after the partial occlusion. Moreover with a mutation rate of 0% the filter cannot trace back the object after it loses it.

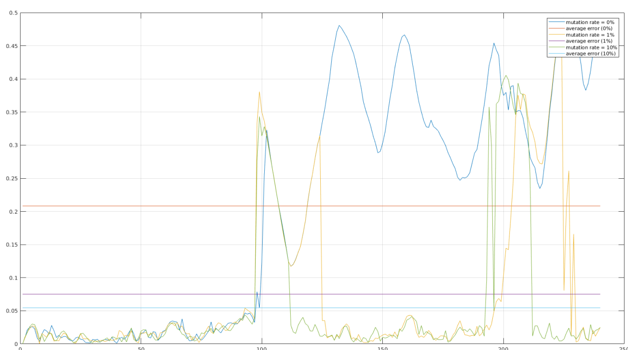

Fig. 14

In figure 15 a filter with the second observation model is used, the blue line has 30 particles while the yellow 50 and the green 100. Naturally the green one performs considerably better than the others trading off accuracy with computation time.

In figure 16 a filter with the first observation model is used, the blue line has 100 particles while the yellow 1000 and the green 10000. Also in this case the higher the number of particles the higher the accuracy at the expenses of computation time. In this graph we can also note how the 100 particles filter diverges after the total occlusion, the 1000 particles filter instead is able to relocate the object but it suffers from the light change while the 10000 particles filter is completely robust to light change.

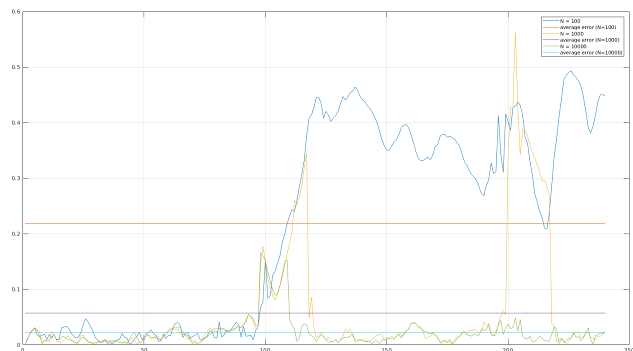

Fig. 16

5 Conclusion

Particle filter is an interesting method to solve the problem of tracking objects in digital videos. It has several advantages with respect to the kalman filter and it can perform decently without analyzing the whole state space. For the digital image tracking several models can be used and the observation part is certainly one of the most important. As we have seen color based models can achieve good results while keeping a low complexity, edge based models are instead more complicated and often less reliable.

5.1 Possible improvements

As possible further ways to improve this estimator a number of algorithm can be added. We did not concentrate much on the resampling step and more advance method such as the annealed particle filter can be considered as in [6]. More cue such as texture based and a support vector machine for the initialization can be considered.

REFERENCES

[1] K. Nummiaro, E. Koller-Meier, L. Van Gool, *An adaptive color-based particle filter*, Image and vision computing 21 (1), 99-110, 2003.

[2] C. Yang, R. Duraiswami, L. Davis, *Fast Multiple Object Tracking via a Hierarchical Particle Filter*, Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1.

[3] L, Mihaylova, P. Brasnett, N. Canagarajah, D. Bull, *Object Tracking by Particle Filtering Techniques in Video Sequences*, Advances and challenges in multisensor data and information processing Volume 8, 260-268, 2007.

[4] M. Isard, A. Blake, *CONDENSATION - Conditional Density Propagation for Visual Tracking*, August 1998, Volume 29, Issue 1, pp 5?28.

[5] J. Czyz, B. Ristic, B. Macq, *A Particle Filter for Joint Detection and Tracking of Color Objects*, Image and Vision Computing, Volume 25, Number 8, August 2007.

[6] J. Deutscher, A. Blake, I. Reid, *Articulated Body Motion Capture by Annealed Particle Filtering*, Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No.PR00662).

[7] S. Thrun, W. Burgard, D. Fox, *Probalistic robotics*, MIT Press, (2005).

[8] M. Isard and A. Blake, ?A mixed-state condensation tracker with automatic model-switching?, In proc. ICCV98, January, 1998, pp. 107 ? 112.

[9] A. Lehuger, P. Lechat and P. Perez, ?An adaptive mixture color model for robust visual tracking?, In Proc. IEEE Int. Conf. on Image Process, October, 2006, pp. 573 ? 576.

[10] K. Nummiaro, E. Koller-Meier, and L. Van Gool, ?A color-based particle filter?, In Proc. First International Workshop on Generative-Model-Based Vision, 2002, pp. 53 ? 60.