Improving Visual Tracking Robustness in Cluttered and Occluded Environments using Particle Filter with Hybrid Resampling

Flavio de Barros Vidal, Diego A. L. Cordoba, Alexandre Zaghetto and Carla M. C. C. Koike
Department of Computer Science, University of Brasilia, Brasilia, Distrito Federal, 70.910-900, Brazil

Keywords: Visual Tracking, Particle Filter, Hybrid Resampling.

Abstract: Occlusions and cluttered environments represent real challenges for visual tracking methods. In order to increase robustness for such situations, we present, in this article, a method for visual tracking using a Particle Filter with Hybrid Resampling. Our approach consists of using a particle filter to estimate the state of the tracked object, and both particles' inertia and update information are used in the resampling stage. The proposed method is tested using a public benchmark and the results are compared with other tracking algorithms. The results show that our approach performs better in cluttered environments, as well as in situations with total or partial occlusions.

1 INTRODUCTION

Visual tracking is an important technique used in many applications (for example mobile, aerial and manipulator robots) working in structured and unstructured environments(Siradjuddin et al., 2013)(Bakhshande and Taghirad, 2013). The process of combining visual tracking and others techniques is widely known and used. The design of systems based on a visual tracking technique presents challenging problems such as in situations with total or partial occlusions and cluttered environments.

The aim of visual tracking is to detect a target and to determine its position and trajectory in a video sequence. Applications in this field are becoming very common (Gao et al., 2012) (Ge et al., 2012) (Leibe et al., 2008), along with the evolution and lower costs of camera and computer technologies.

Visual tracking can be seen as a correspondence subproblem in vision-based motion analysis. The correspondence problem deals with determining the matching between elements of two frames in a sequence. It can, then, be applied for tracking purposes by determining the movement of an entire target region over a long sequence of images. Due to the small spatial and temporal differences between consecutive frames, the correspondence problem can also be stated as the problem of estimating the apparent motion of the image brightness pattern.

The solution of the correspondence problem can roughly follow two strategies: differential methods and window-matching methods. Differential techniques are based on the spatial and temporal variations of the whole image brightness, generating then the optical flow. Methodologies for motion detection based on differential techniques can be modified to perform object tracking in a sequence of images (Vidal and Alcalde, 2005). However, these techniques demand numerical calculation of derivatives that could be impracticable in circumstances where there is a high level of noise, reduced number of frames or the effect of aliasing in the image acquisition process. Window-matching techniques (Anandan, 1989) are based on the assessment of the degree of similarity among regions in sequential images, so that an object may be recognized and its position inferred in subsequent frames. Window-matching techniques can be applied to image tracking and to other issues in computing vision.

Occlusions and cluttered environments represent real challenges for visual tracking methods, because in these conditions the target can no longer be observed. Since obstacles may be treated as nonlinearities, non-linear algorithms such as particle filter are proposed to overcome occlusions and cluttered environments in tracking. A Particle filter is one of many techniques that perform Recursive Bayesian Estimation, and it estimates recursively the posterior density function over a certain state space. Many recent approaches using Particle Filters for visual tracking can be found in those papers of (Romo-Morales et al., 2013), (Limprasert et al., 2013), (Mohan and

de Barros Vidal F., A. L. Cordoba D., Zaghetto A. and M. C. C. Koike C..
Improving Visual Tracking Robustness in Cluttered and Occluded Environments using Particle Filter with Hybrid Resampling.
DOI: 10.5220/0004731006050612
In *Proceedings of the 9th International Conference on Computer Vision Theory and Applications* (VISAPP-2014), pages 605-612
ISBN: 978-989-758-009-3

Wilscy, 2013), (Zhou et al., 2014), (Maier-Hein et al., 2013), (Rui et al., 2013) and many others in a extensive available literature.

In the case of visual tracking, the density function is a representation of the probability of the target position in a frame of an image sequence. The main idea of Particle Filters is to represent the *a posteriori* density function by a set of random samples with associated weights. These associated weights are obtained by a function that reaches its maximum in those samples near the object distinguished features. A major concern regarding Particle Filters is related to the situation where many of its samples drift to low posterior probability regions. The Resampling stage aims to move the set of particles towards regions in state space with higher posterior probability.

In this paper, we propose the use of a particle filter in association with an Hybrid Resampling strategy as a method for robust and accurate response on different tracking scenarios. In Section 2 the Particle Filter methods are introduced and discussed and Section 3 presents the hybrid resampling strategy. In Section 4 the proposed algorithm is applied to two types of visual tracking situations and the results are commented and discussed.

2 THE PARTICLE FILTER

Particle filter is a powerful and flexible estimation technique for nonlinear applications. It is based on simulation and it is usually applied to estimate *Bayesian Models* where all variables are connected in a Markov Chain (Doucet et al., 2001). The main idea is to obtain an approximate representation of the posterior probability density function using a subsequent set of random samples with associated weights.

Let $\{X^i_{0:k}, w^i_k\}^{N_s}_{i=1}$ be a measure that describes a random posterior probability density function (PDF) $p(X_{0:k}|Y_{1:k})$, where $(X^i_{0:k}, i = 0, ..., N_s)$ is a set of support points with associated weights $(w^i_k, i = 0, ..., N_s)$. The state vector $X_{0:k} = (X_j, j = 0, ..., k)$ is the set of all states at time k. The measure vector $Y_{1:k} = (Y_j, j = 1, ..., k)$ is the set of all measures at time k. The weights are normalized by $\sum^{N_s}_{i=1} w_i = 1$ and obtained by Eq. 1,

$$p(X_{0:k}|Y_{1:k}) \approx \sum^{N_s}_{i=1} w^i_k \delta(X_{0:k} - X^i_{0:k}). \tag{1}$$

The theory of Importance Sampling (Doucet et al., 2000) ensures that we can build an estimator if each X^i_j and sample weights are calculated according to Eqs. 2 and 3,

$$X^i_j \propto q(X^i_k|X^i_{k-1}, Y^i_k), \tag{2}$$

$$w^i_k \propto w^i_{k-1} \frac{p(Y_k|X^i_k)p(X_k|X^i_{k-1})}{q(X^i_k|X^i_{k-1}, Y^i_k)}. \tag{3}$$

The distribution $q(X^i_k|X^i_{k-1}, Y^i_k)$ is called *importance density* and a good choice for this distribution can be defined as $q(X^i_k|X^i_{k-1}, Y^i_k) = p(X_k|X^i_{k-1})$. Also, Equation 3 can be reduced to:

$$w^i_k \propto w^i_{k-1} p(Y_k|X^i_k). \tag{4}$$

A common problem in this algorithm is the degeneration effect, as explained in (Arulampalam et al., 2002), and in order to solve it, we can use an effective sample size ($\hat{N}_{eff}$) defined by,

$$\hat{N}_{eff} = \frac{1}{\sum^{N_s}_{i=1} w^i_k}. \tag{5}$$

2.1 Resampling

The resampling process eliminates particles with small weights. These weak particles are replaced by others with higher weights, which defines another set of samples as a better representation for discretized $p(X_k|Y_k)$, described in Eq. 6,

$$p(X_k|Y_{1:k}) \approx \sum^{Ns}_{i=1} w^i_k \delta(X_k - X^i_k). \tag{6}$$

The result of the resampling process is a new set of particles with uniform weight $1/N_s$.

2.2 Color Distribution Model

For visual tracking, a color-based model is used to achieve robustness in situations with non-rigidity, rotation and partial occlusion in image domain. In our approach, we have choosen the HSV color space, due to its better stability under lighting changes compared to RGB. A descriptor based on color histogram (with 10 bins for Hue (H) and Saturation(S) channels) was used as input for the proposed particle filtering scheme.

2.3 Weights Setup

For each generated sample of the input image, the histogram of the tracked region of interest H_i is evaluated. Then, the *Bhattacharyya Distance* (Straka and Šimandl, 2005), $d_{H_O-H_i}$, between H_i and the histogram of the tracked object, H_O, is calculated as described in the Eqs. 7 and 8, is calculated.

$$MB = \sum_i \sqrt{H_N}\sqrt{S_i} \tag{7}$$

$$d_{H_O-H_i} = \sqrt{1-MB} \tag{8}$$

This value is used to calculate the weight of each particle according to Eq. 9,

$$w_i = exp(-\lambda d^2_{H_O-H_i}), \tag{9}$$

where λ is equal to 20 (more detailed in (Prez et al., 2002)) and $d_{H_O-H_i}$ is the value of Bhattacharyya distance for the sample i. The above equation can assure that if the samples have a high similarity with the target's histogram, the weights are adjusted to large values. If the similarity is small, the weights are reduced to small values.

2.4 Updating Model

To update the target model, we use the average weight of the particles that are close to the tracked object. If this value is above a fixed threshold, w_{min}, then the histogram of the tracked object is updated. Equation 10 shows how our approach prevents wrong update of the histogram values ($H_{O:k}$) with undesired color values of the tracked region surroundings (as proposed (Li and Chua, 2003)).

$$\begin{aligned} H_{O:k} \;\; = \;\; & (1-\alpha-\beta-\gamma)H_{O:k-1} + \\ & +(\alpha)H_{\Sigma w_i.X} + (\beta)H_{MP} + (\gamma)H_{O:k=0} \end{aligned} \tag{10}$$

The values of α, β and γ are chosen according to changes in the target estimation: α is the normalized weight of the particles estimation, β is proportional to the normalized weight of the highest particle value and γ is defined as 0.1.

3 THE PARTICLE FILTER WITH HYBRID RESAMPLING

In order to overcome the problems related to occlusions, we used the basic structure of a particle filter based on Sequential Importance Resampling(SIR) (Gordon et al., 1993), which is described in the following sections.

3.1 Resampling Stage

Here we use a modified version of the approach proposed by (Kitagawa, 1996) and detailed in Algorithm 1. The modified version of SIR Particle Filter algorithm is shown in Algorithm 2. The tracking process initiates with the assumption that there is no target occlusion and the proposed particle filter uses the discrete dynamic state model described in Eq. 11,

$$\begin{aligned} X'_k &= X_{k-1} + r_k \\ X''_k &= X'_{k-1} + I_{k-1} \\ Y_k &= h(X_k, s_k). \end{aligned} \tag{11}$$

Algorithm 1: Resampling Algorithm.

Data: $[\{X_k^{j*}, w_k^j, i^j\}_{j=1}^{N_s}]$
Result: $[\{X_k^i, w_k^i\}_{i=1}^{N_s}]$
Initialization PDF $c_1 = 0$;
for $i = 2 : N_s$ **do**
    Build PDF: $c_i = c_{i-1} + w_k^i$
end for
Random initialization: $u_1 \sim U[0, N_s^{-1}]$ **for**
$j = 1 : N_s$ **do**
    Move along the PDF: $u_j = u_1 + N_s^{-1}(j-1)$;
    while $u_j > c_i$ **do**
        $i = i + 1$
    end while
    Assign sample $X_k^{j*} = X_k^i$;
    Assign weight $w_k^j = N_s^{-1}$;
    Assign parent $i^j = i$;
end for

Algorithm 2: Particle Filter with Hybrid Resampling.

Data: $[\{X_{i:k-1}, w_{i:k-1}\}_{j=1}^{N_s}, Y_k]$
Result: $[\{X_{i:k}, w_{i:k}\}_{i=1}^{N_s}]$
initialization;
for $i = 1 : N_s$ **do**
    $X_{i:k} \sim p(X_k|X_{i:k-1})$;
    Calculate $w^*_{i:k} = p(Y_k|X_{i:k})$ (Eq. 9);
end for
for $i = 1 : N_s$ **do**
    if $1 < \frac{wi*}{wi}$ **then**
        $X = X^*$ (Eq.11);
        $w_i = w_i^*$;
    else
        $X = X_{k-1}$ (Eq.11);
        $w_i = w_{i:k-1}$;
    end if
end for
Calculate $\bar{v}$ (Eq.12);
for $j = 1 : N_s$ **do**
    Normalization $w_{i:k} = \frac{w_{i:k}}{\sum i = 1^{N_s} w_{i:k}}$;
end for
Calculate $\hat{N}_{eff}$ (Eq.5);
if $\hat{N}_{eff} < N_{lim}$ **then**
    Resampling Algorithm 1;
end if
Update Histogram target (Eq. 10);
Estimate $\hat{X}$ (Eq.18);

The state vector X_k estimates the position (vertical and horizontal) of the target in the image domain, obtained from a rectangle that encloses the tracked object. The state vector X'_k and X''_k are derivatives (first and second order respectively) of the target position.The random variables r_k and s_k are mutually independent, modeled by Gaussian functions, and they describe the process and measuring noises respectively. $h(.)$ is a measure state function. I_{k-1} is the inertial factor, responsible for providing the inertial movement of the samples and obtained from a Gaussian distribution weighted by the velocity of the particles.

Assuming that the tracked object velocity v between frames is uniform, it may be evaluated as,

$$v_i = (X'_{i:k} - X'_{i:k-1}). \tag{12}$$

The expressions of transition probabilities are defined by Eqs. 13, 14 and 15 respectively.

$$\hat{p}(X_k|Y_{k:1}) = argmax\{\pi(X_{i:k})\}, \tag{13}$$

$$p(X_k|X_{k-1}) = p(X_{k-1}|r_k, I) and \tag{14}$$

$$p(Y_k|X_k) = N(Y_k|h(X_k), s_k). \tag{15}$$

$p(Y_k|X_k)$ is a Normal distribution ($N(.)$), $\pi(X_{i:k})$ is the *a posteriori* distribution from state samples X, defined by Eq. 16 and restricted by Eq. 17.

$$\pi(X) = p(\mathbf{Y}_k|\mathbf{X}_{i:k})\,[\Omega(.)], \tag{16}$$

$$\Omega(.) = \sum_{i=1}^{\lambda} p(\mathbf{X'}_k|\mathbf{X'}_{i:k-1}) + \sum_{i=\lambda}^{N_s} p(\mathbf{X''}_k|\mathbf{X''}_{i:k-1}), \tag{17}$$

where λ is a part of overall particles set and N_s is the maximum number of particles in the filter.

When the target can not be observed in a frame (for example because of an occlusion), the state transition for each set of particles is modified. The first group (first sum of Eq.17) changes from a normal distribution to an uniform distribution around the last estimation before the occlusion, as described in Eq. 18,

$$X'_i = \hat{X}_{i:k-1} + r_k, \tag{18}$$

where the state vector $\hat{X}_{k-1}$ describes the time before occlusion and r_n shows the value evaluated from a uniform distribution $U(u|l_k, u_k)$. The lower and upper limits changes during the remaining frames. The other state vector X'' behaves according to the latest update of the estimated velocity state vector X', before occlusion. Following these constraints an estimation of the object position is possible, even though total occlusion or high cluttered environments occurs. When the missing target reappears, or the occlusion is over, the particles are updated by the state transition shown in Equation 11.

4 EXPERIMENTAL RESULTS

In order to evaluate the proposal method, we used the *Bonn Benchmark on Tracking - Bobot* [1]. The *Bobot* dataset includes several sequences with many types of tracking objects such as people, mugs, etc. and a complete ground-truth (GT) with spatial positions (horizontal and vertical) and size (height and weight) of each tracked object. All available sequences have an spatial resolution 320×250 pixels and frame rate of 25 fps. The proposed Particle Filter with Hybrid Resampling (PFHR), described in Section 3, uses up to 200 particles. We compared the proposed method with two other algorithms: (a) a deterministic algorithm based on template matching (WM); and (b) a basic implementation of the SIR Particle Filter (also setup up to 200 particles).

Our method was run in more than six image sequences scenarios from the above mentioned benchmark. Here we show illustrative results for two sequences from the *Bobot Database* (additional results in Table 1). The first sequence presents many characteristics of cluttered environments (Subsection 4.1). The second sequence shows an outdoor sequence where several objects present high similarity with the tracked object and total occlusion happens in more the one occasion (Subsection 4.2).

4.1 Cluttered Environments Sequence

This sequence presents many abrupt background changes, camera motion and scale transformation. Figure 1 shows some frames from the sequence with cluttered environments.

Many errors occur in *WM* algorithm during the tracking of the blue mug (See Figs. 1 and 2), especially when similar objects appear at the bottom of the image. The basic SIR Particle Filter algorithm has an adequate response with respect to the spatial position of tracked object, but the object size is incorrectly estimated. On the other hand, PFHR does a much better job estimating both features, the position and the size of the tracked objects.

4.2 Outdoors with Total Occlusion Sequence

This sequence shows a person (tracked object) walking outdoors while several others people cross the way, generating occlusions (Fig. 3). Besides occlusions, the tracked person performs scale, rotation and translation changes during the movement.

[1] Available in http://www.iai.uni-bonn.de/∼kleind/tracking/

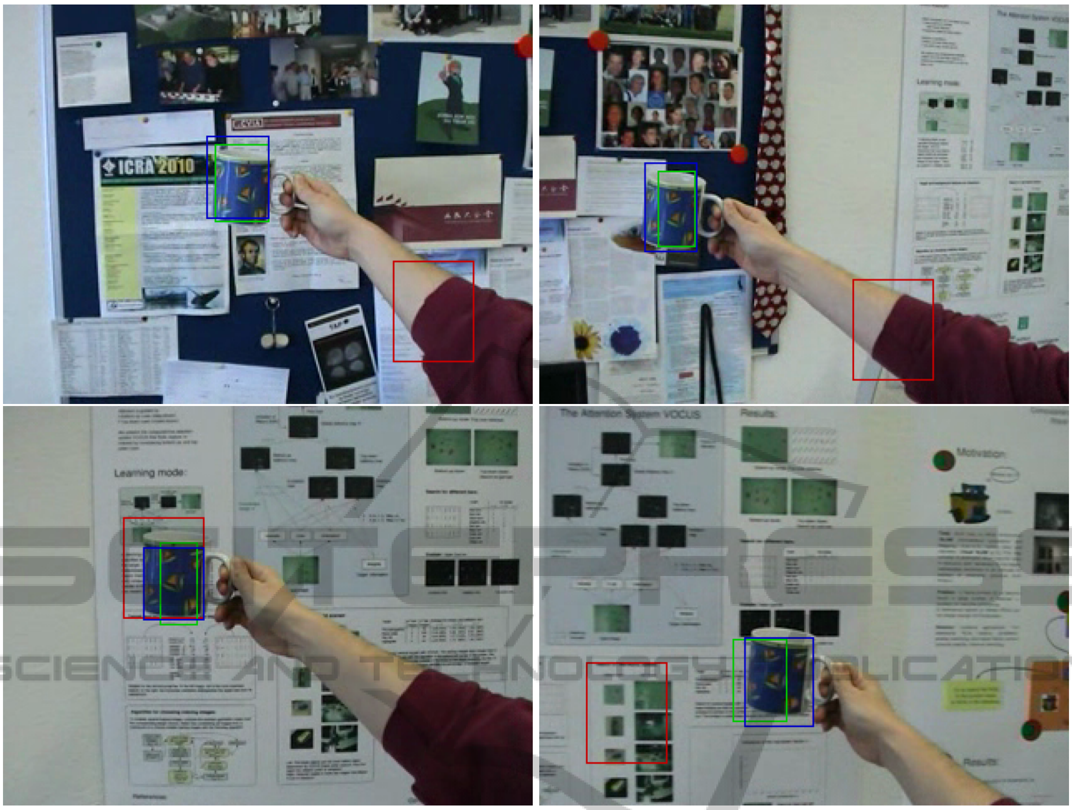
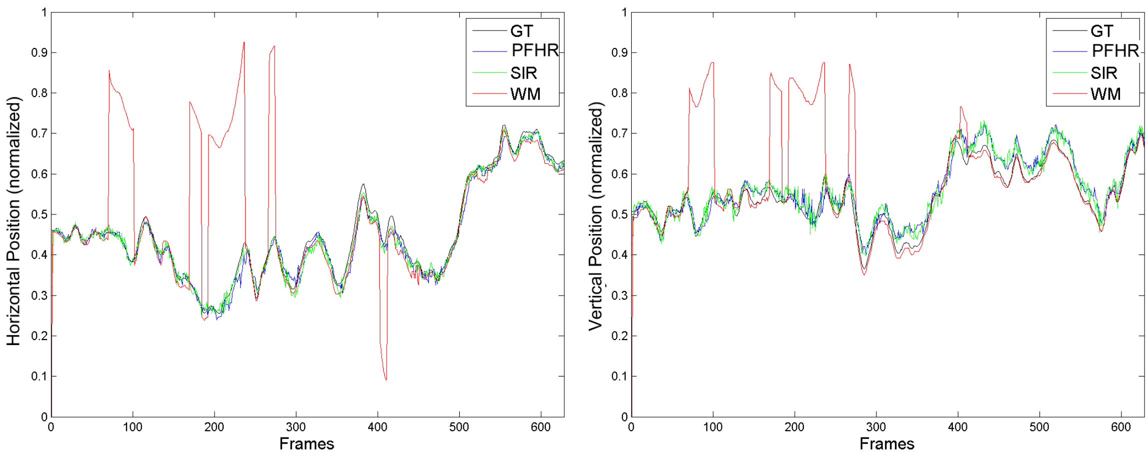


Figure 1: Tracking in cluttered environments. Legend: WM, SIR Particle Filter, PFHR.



(a) (b)

Figure 2: Cluttered environment Sequence - Positions normalized Horizontal (a) and Vertical (b). Ground-truth (GT), WM, SIR Particle Filter, PFHR.

The WM algorithm missed the tracked person after occlusion and in the presence of objects that are similar to the target[2]. The SIR Particle Filter also missed the target after occlusions and it was not able to properly estimate the size of the target. The PFHR algorithm was capable of estimating both the correct position and size of tracked person. While a persistent occlusion occurs, the PFHR strategy performs *a posteriori* distribution of the state vector X from Equa-

[2]For Bobot Benchmark when an occlusion occurs the value assigned by the ground-truth to the horizontal and vertical positions is zero, respectively.

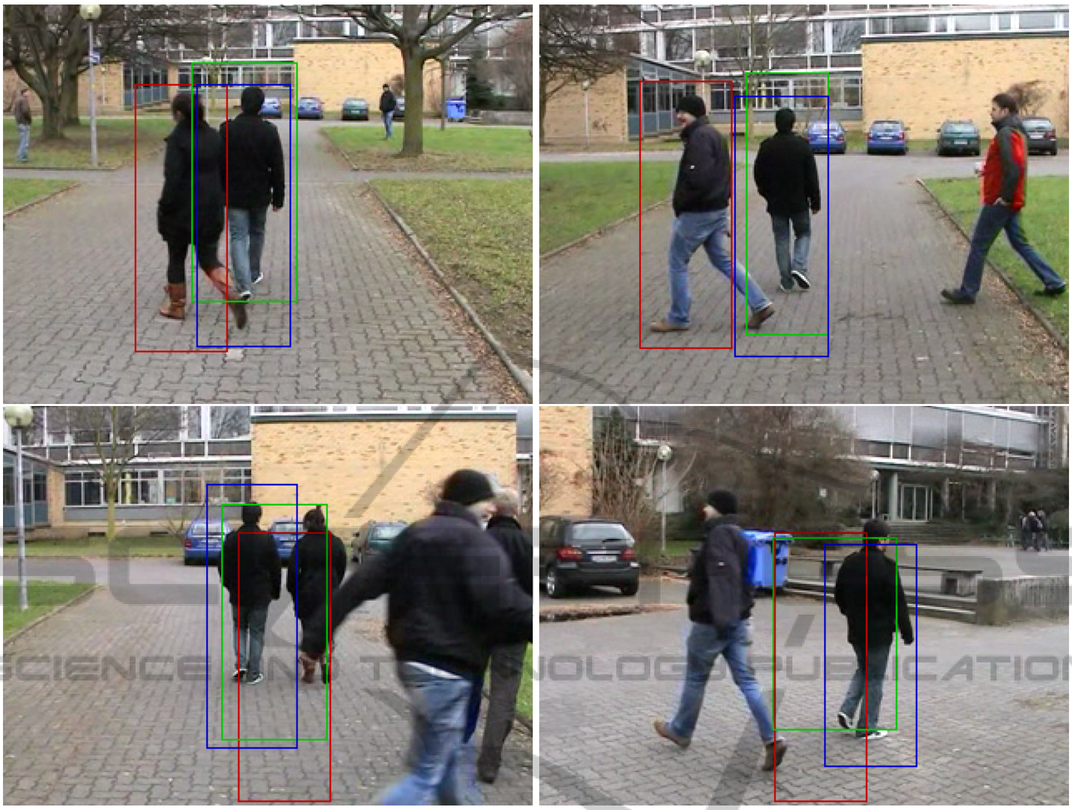


Figure 3: Sequence of the total occlusion of a person walking in outdoor environment. Legend: WM, SIR Particle Filter, PFHR.

tion 17, including values from the dynamic movement model in image domain (Equation 11). Specifically for this occlusion situation we setup half of the number of particles to the first order model and second order model respectively. These values were chosen empirically to obtain the best result for the tracking performed.

As can be seen in the sequence shown in Fig. 3, WM is not able to track the person. The SIR Particle Filter algorithm can track correctly, but with large displacements positions (horizontal and vertical) variations along the frames, as shown in Fig. 4. The PFHR track the person with small errors and after occlusion occurs, even with small oscillations, the algorithm can recover the target after few frames.

4.3 Tracking Performance Evaluation

To adequately evaluate the tracking algorithm, we must also take the estimation of target size into account, as proposed by (Yin et al., 2007). It consists on measuring the overlap between ground truth and estimated areas, as defined by Eq. 19,

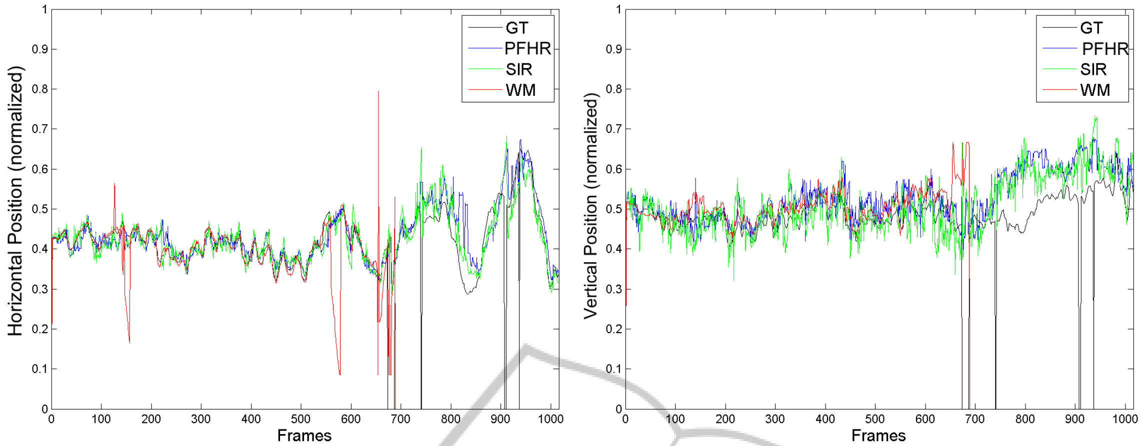
$$A(GT,ST) = \frac{Area(GT \cap ST)}{Area(GT \cup ST)}, \qquad (19)$$

where GT represents the *ground truth* area and ST represents the area estimated obtained the tracking algorithm. In according to (Yin et al., 2007), if $A(GT,ST)$ is greater than threshold, T_{lim}, chosen to be at least 20%, then we have a true positive. Table 1 shows the results (in percentage) of true positives along of the total number of frames where the target is detected, for all method tests. If can be seen that PFHR shows a higher percentage than WM and SIR, for all tested sequences.

4.4 Computational Complexity Analysis

The evaluation of the computational complexity in (Cormen et al., 2001) assigns for each line in the algorithm a variable that represents the running time, regardless the type of the variable (float, integer, ...). If the statement is inside a repetition structure (*while*, *for*, ...), the complexity increases proportionally to the nested loops. Considering n to be the standard number of input particles of the analyzed Particle Filter, its computational complexity is O(n), with no relation to the number of samples.

Comparing the computational complexity obtained from the standard Particle Filter SIR and the al-



(a) (b)

Figure 4: Outdoor occlusion sequence - Positions normalized Horizontal (a) and Vertical (b). Ground truth (GT), WM, SIR Particle Filter, PFHR.

Table 1: Percentage of true positive data.

	PFHR	SIR	WM
Cluttered Environments	**97.60**	95.21	82.93
Outdoor w/ Total Occlusion	**93.79**	90.18	70.07
Background changes	**97.60**	95,21	82,93
Scale variations	92,30	**94.96**	33.90
Trajectory changes	**95.67**	76.87	87.18
Indoor w/ Total Occlusion	**88.93**	86.06	30.08
Overall Average	**94.75**	89.74	64.51

gorithm proposed here (PFHR), we achieve the same computational complexity, even when the resampling methodology described in Algorithm 1 is applied.

5 CONCLUSIONS

In this paper we presented a new approach for visual tracking, that uses a Particle Filter with Hybrid Resampling strategy in order to improve robustness. All tests show that the proposed algorithm (PFHR) active better results than a deterministic algorithm based on template matching, and a basic implementation of the SIR Particle Filter, especially in occlusions and cluttered environments. The algorithm proposed (PFHR) achieve better results, compared to the classical techniques of visual tracking (WM and SIR), especially in occlusions and cluttered environments.

The presented approach would provide improvements for visual tracking due to the fact that the tracking is independent of the motion type (for example random trajectories) and of the object shape. The algorithm also offers flexibility in situations where there is no previous informations about the object to be tracked. Further works may include the implementation of the proposed algorithm on a high level programming language in order to enable its operation in real time scenarios (including timing analysis) and also perform more comparisons with the latest techniques available in visual tracking literature.

REFERENCES

Anandan, P. (1989). A computational framework and an algorithm for the measurement of visual motion. In *International Journal of Computer Vision*, volume 2, pages 283–310.

Arulampalam, M. S., Maskell, S., Gordon, N., and Clapp, T. (2002). A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Transactions on signal procesing*, 50.

Bakhshande, F. and Taghirad, H. (2013). Visual tracking in four degrees of freedom using kernel projected measurement. In *Robotics and Mechatronics (ICRoM), 2013 First RSI/ISM International Conference on*, pages 425–430.

Cormen, T. H., Stein, C., Rivest, R. L., and Leiserson, C. E. (2001). *Introduction to Algorithms*. McGraw-Hill Higher Education, 2nd edition.

Doucet, A., De Freitas, N., and Gordon, N., editors (2001). *Sequential Monte Carlo methods in practice*.

Doucet, A., Godsill, S., and Andrieu, C. (2000). On sequential monte carlo methods for bayesian filtering. *Statistics and Computing*, 10:197–208.

Gao, X., You, D., and Katayama, S. (2012). Seam tracking monitoring based on adaptive kalman filter embedded elman neural network during high-power fiber laser welding. *Industrial Electronics, IEEE Transactions on*, 59(11):4315–4325.

Ge, W., Collins, R., and Ruback, R. (2012). Vision-based analysis of small groups in pedestrian crowds. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(5):1003–1016.

Gordon, N., Salmond, D., and Smith, A. F. M. (1993). Novel approach to nonlinear/non-gaussian bayesian state estimation. *Radar and Signal Processing, IEE Proceedings F*, 140(2):107–113.

Kitagawa, G. (1996). Monte carlo filter and smoother for non-gaussian nonlinear state space models. *Journal of Computational and Graphical Statistics*, 5:1–25.

Leibe, B., Schindler, K., Cornelis, N., and Van Gool, L. (2008). Coupled object detection and tracking from static cameras and moving vehicles. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(10):1683–1698.

Li, J. and Chua, C.-S. (2003). Transductive inference for color-based particle filter tracking. 3:III–949–52 vol.2.

Limprasert, W., Wallace, A., and Michaelson, G. (2013). Real-time people tracking in a camera network. *Emerging and Selected Topics in Circuits and Systems, IEEE Journal on*, 3(2):263–271.

Maier-Hein, L., Mountney, P., Bartoli, A., Elhawary, H., Elson, D., Groch, A., Kolb, A., Rodrigues, M., Sorger, J., Speidel, S., and Stoyanov, D. (2013). Optical techniques for 3d surface reconstruction in computer-assisted laparoscopic surgery. *Medical Image Analysis*, 17(8):974 – 996.

Mohan, K. and Wilscy, M. (2013). Object ranging and tracking for aircraft landing system. In *Signal Processing Image Processing Pattern Recognition (ICSIPR), 2013 International Conference on*, pages 278–282.

Prez, P., Hue, C., Vermaak, J., and Gangnet, M. (2002). Color-based probabilistic tracking. pages 661–675.

Romo-Morales, L., Sanchez, A., Parra-Vega, V., Garcia, O., and Ruiz-Sanchez, F. (2013). Visual control for trajectory tracking of quadrotors and real-time analysis on an emulated environment. In *American Control Conference (ACC), 2013*, pages 6315–6321.

Rui, T., Zhang, Q., Zhou, Y., and Xing, J. (2013). Object tracking using particle filter in the wavelet subspace. *Neurocomputing*, 119(0):125 – 130. ¡ce:title¿Intelligent Processing Techniques for Semantic-based Image and Video Retrieval¡/ce:title¿.

Siradjuddin, I., Behera, L., McGinnity, T., and Coleman, S. (2013). Image-based visual servoing of a 7-dof robot manipulator using an adaptive distributed fuzzy pd controller.

Straka, O. and Šimandl, M. (2005). *Using the Bhattacharyya distance in functional sampling density of particle filter*, pages 1–6. IFAC, Prague.

Vidal, F. B. and Alcalde, V. H. C. (2005). Motion segmentation in sequential images based on the differential optical flow. *2nd International Conference on Informatics in Control, Automation and Robotics -ICINCO*, pages 94–100.

Yin, F., Makris, D., and Velastin, S. (2007). Performance evaluation of object tracking algorithms. *Proceeding Tenth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*.

Zhou, X., Li, Y., and He, B. (2014). Entropy distribution and coverage rate-based birth intensity estimation in gm-phd filter for multi-target visual tracking. *Signal Processing*, 94(0):650 – 660.