

Robust Object Tracking in Infrared Video via Particle Filters

Edgardo Comas*, Adrián Stácul* and Claudio Delrieux+

* *Instituto de Investigaciones Científicas y Técnicas para la Defensa, San Juan Bautista de La Salle 4397 - Villa Martelli, Buenos Aires, Argentina.*

* *Universidad Tecnológica Nacional, Facultad Regional Buenos Aires, Medrano 951, Buenos Aires, Argentina.*

+ *Universidad Nacional de Sur, Laboratorio de Ciencias de las Imágenes, and CONICET, Avenida Colón 80 - Bahía Blanca, Argentina.*

Received 05th of September 2019; accepted 7th of February

Abstract

In this paper we investigate the effectiveness of particle filters for object tracking in infrared videos. Once the user identifies the target object to be followed in position and size, its most representative feature points are obtained by means of the SURF algorithm. A particle filter is initialized with these feature points, and the location of the object within the video frames is determined by the average value of the particles that have a greater similarity with the target. Our aim is to make possible unupervised object tracking in unmanned night flights. Two different field tests were carried out to study the filter behaviour in comparison with previously used methods in the bibliography. The first one was tracking an unmanned aerial vehicle (UAV) in the open. The second one was to identify a heliport in a noisy infrared zenithal video take. In the first test, the UAV was followed by another positioning system simultaneously, thus allowing the comparison of both systems, and the evaluation in the improvement introduced by the particle algorithm.

Key Words: Image Analysis, Pattern Recognition, Tracking, Particle Filter

1 Introduction

Target detection and tracking from infrared videos has several applications in the fields of early warning, precision guidance, and surveillance, among others [1] [2]. Current research in the subject, in which particle filters were applied, are mostly focused daylight and visible light takes, on small targets, noise-free background, or with very slow variations [3] [4] [5]. To adequately perform object tracking in infrared image sequences, different algorithms are embedded in the particle filter framework, such as histogram modification and mean shift [3]. Fuzzy clustering [6] [7] combines border extraction filters applied in combination with fuzzy logic and clustering. To address fast target movements and appearance changes, an eigenspace model is used combined with a saliency method in a two-step sampling and observation framework [4] [8].

Particle filters are a special kind of sequential Monte Carlo methods, which are specifically advantageous in transition state models and measurements, especially when these are non-linear and not Gaussian [9]. The observations update the probability of validating a hypothesis, that is, they use all the available information to reduce the amount of uncertainty present in an inference or decision problem, obtaining in a sequential

Correspondence to: ecomas@frba.utn.edu.ar

Recommended for acceptance by Mikhail Morezov

<http://doi.org/10.5565/rev/elcvia.1185>

ELCVIA ISSN: 1577-5097

Published by Computer Vision Center / Universitat Autònoma de Barcelona, Barcelona, Spain

way a numerical approximation [10]. Monte Carlo computational techniques [11] were developed to represent posterior probability densities in terms of random samples with their associated weights. In particular, the particle estimators do not have the restrictive hypothesis of the Kalman estimator, and therefore they can be applied to non-linear models with non-Gaussian and multimodal noise, in which the reliable numerical estimate is a function of an adequate number of samples [12].

In this work, we expect to extend the applicability of particle filters for object tracking in infrared image sequences, making feasible unmanned night flights in the aforementioned contexts. We propose an improvement of the target image based on a reduction of the background incidence. This is achieved by means of a modification of the mean intensity value and its histogram. After applying this process, the SURF algorithm is used to detect characteristic points of each particle and, with them, a similarity measure with respect to the reference particle. To cope with unpredictable movements of the target and camera variations, in the resampling process a dispersion value is considered as a function of the number of valid particles, so that the area covered during tracking is dynamically modified.

2 Particle filter tracking

Particle filter tracking is performed combining new particle information with the one obtained previously using Bayes theorem [11]. To define the estimation problem, consider the evolution model of the state X of a system by the following equation:

$$X_k = f(X_{k-1}, v_{k-1}), \quad (1)$$

where the state vector $X \in R^n$ contains all the variables that will be dynamically estimated. These variables evolve according to the structural equation (1), where f is the non-linear function of the state variables, and $v \in R^n$ represents the state of the noise system. On the other hand, consider the observation model corresponding to the measurements $Z \in R^n$ available in t_k , with $k = 1, 2, \dots$. These measurements are related to the state variables X by means of a non-linear function h in the following equation:

$$Z_k = h(X_k, n_k), \quad (2)$$

where $n \in R^N$ is the measurement noise, and h is known as an observation model. The estimation problem focuses on obtaining state information X_k based on the evolution model (1) and the measurements $Z_{1:k} = \{Z_i, k = 1, \dots, k\}$ given by the observation model (2). The evolution-observation model given by the equations (1) and (2) are based on the following hypotheses that are referring to the following sequences (where $P\left(\frac{a}{b}\right)$ denotes the probability of event a given observation b):

a) $X_k, k = 1, 2, \dots$ is a Markov process [10] [13]

$$P(X_k | X_0, X_1, \dots, X_{k-1}) = P(X_k | X_{k-1}),$$

b) $Z_k, k = 1, 2, \dots$ is a Markov process regarding the historical data of X such that

$$P(Z_k | X_0, X_1, \dots, X_k) = P(Z_k | X_k),$$

c) and the sequence of past observations only depends on its history, that is

$$P(X_k | X_{k-1}, Z_{k-1}) = P(X_k | X_{k-1}).$$

In addition to the evolution-observation model given by the equations (1) and (2), it is necessary to consider that for all $i \neq j$, the noise vectors $v_i \wedge v_j$ as well as the vectors $n_i \wedge n_j$ are mutually independent of the initial state X_0 and also mutually independent for all $i \wedge j$. Assuming that we know $P(Z_0) = P(X_0)$, then the probability density $P(Z_{1:k})$ is obtained by means of the Bayesian estimator in two steps, prediction and updating [12] [13].

The principle of the particle filter is to represent the posterior probability density function as a set of random samples (particles) with their associated weights. After that, it is possible to calculate an estimation of the most probable particle based on these samples and their weights. As the number of samples increases, the Monte Carlo characterization approaches the representation of the a posteriori probability function, and the solution approaches an optimal Bayesian estimator [14].

Let us take the set of states up to t_k , $X_{0:k} = \{X_j, j = 0, \dots, k\}$, the set of particles $\{Z_{0:k}^i, i = 0, \dots, N\}$ and their associated weights $\{W_k^i, i = 0, \dots, N\}$, where N is the total number of particles, and the weights are normalized such that

$$\sum_{i=1}^N W_k^i = 1.$$

Then, the a posteriori density in t_k can be approximated by:

$$P(X_{0:k}|Z_{1:k}) \approx \sum_{i=1}^N W_k^i \delta(X_{0:k} - X_{0:k}^i), \quad (3)$$

where $\delta(\cdot)$ is Dirac's delta function, and considering the hypotheses corresponding to the expressions (1) and (2) the density a posteriori (3) can be written as [15]:

$$P(X_k|Z_{1:k}) \approx \sum_{i=1}^N W_k^i \delta(X_k - X_k^i). \quad (4)$$

A common problem in this algorithm is the phenomenon of degeneration, where after a few states, all but a few particles (usually one) will have an insignificant weight [12] [15]. This can be solved resampling the particles. However this creates another problem, which is the increasing information uncertainty arising in the random sampling process [15]. This resampling can be done in two ways, the first one is known as "Sampling Importance Resampling" (SIR), where the resampling is done in each cycle of the algorithm, and the other alternative is to perform the resampling each time it is necessary. That is when the effective number of particles reaches a minimum threshold number; this is the so-called "Sampling Importance Sampling" (SIS).

3 The SURF Algorithm

The Speeded-up Robust Features SURF algorithm is a robust detector of characteristic points in images. These points are invariant to scale, rotation, translation, and partially to illumination changes. It is based on an enhancement of the "Scale Invariant Feature Transform" SIFT algorithm [16] producing fewer features per detected point at a much faster rate.

The SURF algorithm uses integral images, which accelerates the computation of representations at different scales of the original image according to their differential features. Integral images accelerate the computation of second order filters by means of Haar wavelets. The application of the Hessian differential operator matrix at different scales allows determining key points and their robust descriptor features with a good performance in precision and execution time [17] [10].

Given an image I , and a point $X = (x, y)$ in this image, the Hessian matrix $H_{(X,\sigma)}$ in $X = (x, y)$ to the scale σ is defined as:

$$H_{(X,\sigma)} = \begin{bmatrix} L_{x,x,(X,\sigma)} & L_{y,x,(X,\sigma)} \\ L_{x,y,(X,\sigma)} & L_{y,y,(X,\sigma)} \end{bmatrix},$$

where $L_{x,x,(X,\sigma)}$, $L_{x,y,(X,\sigma)}$, $L_{y,x,(X,\sigma)}$ and $L_{y,y,(X,\sigma)}$ represent the convolution product of the second derivative of the Gaussian $\frac{\partial^2}{\partial X^2} g_{(X,\sigma)}$ with the Image I in the point $X = (x, y)$ [16], see equation (5).

$$L_{x,x,(X,\sigma)} = \frac{\partial^2}{\partial X^2} * [G_{(x,y,\sigma)} * I_{(x,y)}]$$

$$L_{x,x,(X,\sigma)} = \left[\frac{\partial^2}{\partial X^2} * G_{(x,y,\sigma)} \right] * I_{(x,y)} \quad (5)$$

$$\wedge G_{(x,y,\sigma)} = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

The determinant of the Hessian matrix allows the calculation of the scale of the point, defined as follows:

$$|H_{(X,\sigma)}| = D_{x,x}D_{y,y} - (\omega D_{x,y})^2,$$

where $D_{x,x}$, $D_{y,y}$, and $D_{x,y} = D_{y,x}$ are the approximations of the partial derivatives, and ω is the balance factor of the determinant [16], obtained from equation (6) where $|\cdot|_F$ is the Frobenius norm [18], see equation (7).

$$w = \frac{|L_{x,y,(X,\sigma)}|_F |D_{y,y}|_F}{|L_{y,y,(X,\sigma)}|_F |D_{x,x}|_F} \quad (6)$$

$$|A|_F = \sqrt{\sum_{i=1}^m \sum_{j=1}^n (a_{i,j})^2} \quad (7)$$

Subsequently, the application of the Haar-Wavelet filters in a circular region with a radius of 6σ , provide a set of outputs in horizontal and vertical directions, (dx and dy respectively), and the dominant orientation are obtained as the sum of all responses inside a sliding window of $\frac{\pi}{3}$ [16].

To obtain the feature descriptors for a certain scale and for each characteristic point, a rectangular region centered on the point with size of $20\sigma \times 20\sigma$ is constructed in the dominant orientation and it is divided into four sub-regions of 4×4 , and for each sub-region the Haar-Wavelet is applied obtaining the horizontal and vertical responses d_x and d_y respectively, a characteristic vector $V = (\sum d_x, \sum d_y, \sum |d_x|, \sum |d_y|)$ is formed. Then for each point a total of 64 SURF descriptors are generated [16] [17].

4 Background subtraction

To improve target detection, background subtraction was performed. Background image subtraction techniques are especially feasible in applications where the camera has a constant focal distance, with no

translations or rotations. In this way the background is almost stationary and the target object moves in the foreground [19].

In our case, the situation is approximately similar since we have a camera in motion together with the target object to be tracked. Another substantial difference is that the background image usually remains static, but that is not the case in our context. The object to be tracked appears behind a group of trees, and its leaves are in motion due to the wind.

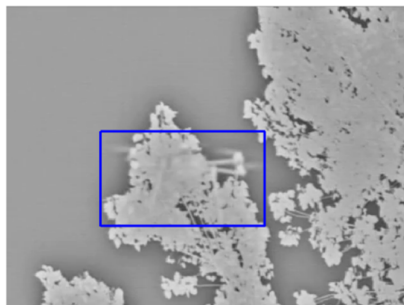


Fig. 1. Frame of the original acquired infrared video

In Fig. 1 we show a frame of the original infrared video, where the UAV to be tracked appears behind a group of trees. Considering that the object target can be masked behind the trees to avoid detection, background suppression algorithm was used.

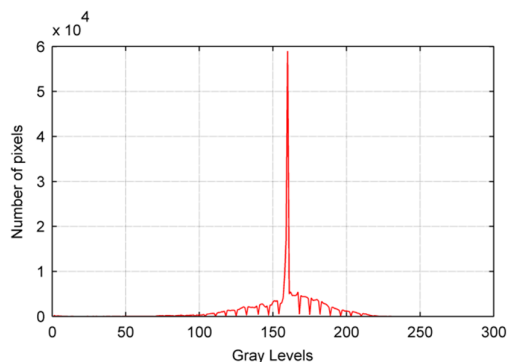


Fig. 2. Histogram of the original acquired infrared video

In Fig. 2 we show the histogram of the image of Fig. 1. In this histogram it can be seen that the image has very low dynamic range in intensity. This causes the SURF algorithm to have a poor performance, and very few characteristic points are detected, if at all. To enhance the response of the SURF algorithm, a subtraction of the average of the image [20] together with an increase in dynamic range was used [21] [22].

Given that the UAV's temperature is higher than the temperature of the trees and the sky in the background, the subtraction of the average value of the image does not produce any significant loss of information. An improvement in dynamic range can be achieved multiplying the result of the previous subtraction by a proportional factor to the relationship between the standard deviation of the image (as measurement of the contrast [21]) divided by the maximum value of the luminosity, (see equation 8), where I and I_p are the input image and processed image, N and M are the number of pixels X and Y respectively, $\sigma(I)$ and $\max(I)$ the standard deviation and the maximum value of image.

$$I_p = \left[I - \alpha \sum_{i=1, j=1}^{N, M} I_{(i, j)} \right] \beta \frac{\sigma(I)}{\max(I)} \quad (8),$$

The adaptive factors α and β are obtained experimentally and with values of $\alpha = 0.84314$ and $\beta = 25$ we can see in Fig. 3 the improvements in the image of Fig. 1. In Fig. 4 we see its histogram of the image of Fig. 3, where the contrast and the dynamic range of the luminosity intensity were improved.

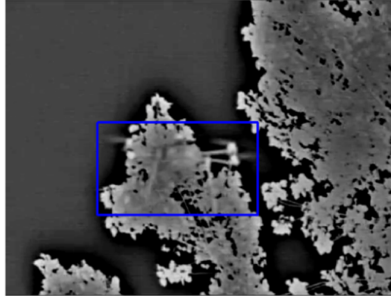


Fig. 3. The same frame as in Fig. 1 after background subtraction.

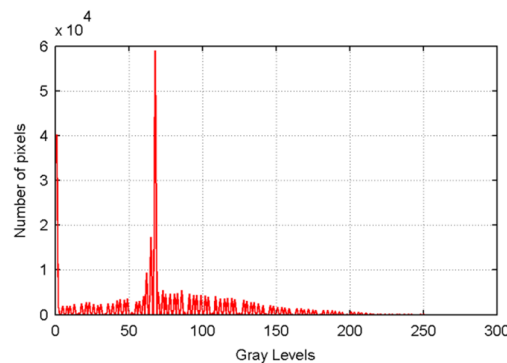


Fig. 4. Histogram of the image of Fig. 3

After the background suppression, the SURF algorithm applied to the rectangle area obtained more than 200 characteristic points, which is adequate considering that the particle filter works well with a maximum of 50 characteristics points per particle.

5 Particle filter algorithm

Basically our method deploys a series of random particles within the space of states, each with its associated weight. These particles are possible states of the process (in this case, the target position), and their weights represent their a posteriori probability of the density function as estimated from the observations. In the case of its application to the tracking objects in video images, we can describe each of its steps in the algorithm of Fig. 5.

A) Initialization: In this stage all the initial conditions of the filter and the parameters that define its behaviour are defined, such as number of particles, number of octaves, metrics and scale levels for the SURF algorithm, maximum number of characteristic points to analyse, metrics of the acceptance threshold level for considering acceptable a point, and probability threshold level for particle acceptance.

B) Definition of the object to be followed: through the intervention of the user, the target object to be followed or reference object is defined, which is characterized by its position, size and its speed, described by the following state vector $(x, y, l_x, l_y, v_x, v_y, v_{lx}, v_{ly})$.

C) Obtaining the characteristic points: applying the SURF algorithm, the characteristic points of the target object are obtained, see (Fig. 6), where the number of points was defined in the initialization step.

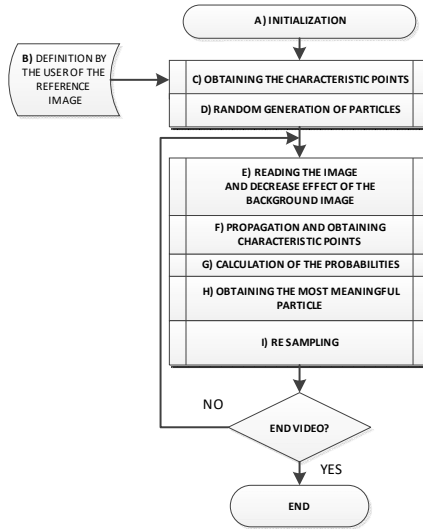


Fig. 5. Particle filter algorithm

D) Random particle generation: The particles of the filter are initialized randomly throughout the screen, with their respective probabilities, and their positions are bound within the video frame (see Fig. 7).

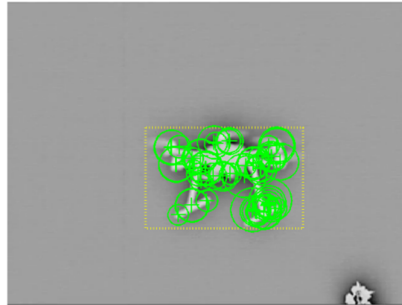


Fig. 6. SURF characteristics point

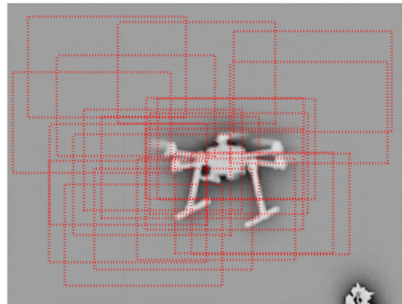


Fig. 7. Initial particles distributions

E) Frame acquisition, particle propagation, and new characteristic points: The new frame is acquired, and then the characteristic points are computed for each particle following step C). For the particle propagation, the following state evolution model is applied, in correspondence with equation (1):

$$P_k = AP_{k-1} + Av_{k-1}, \quad (9)$$

$$A = \begin{bmatrix} I_{4 \times 4} & \Delta t \cdot I_{4 \times 4} \\ 0_{4 \times 4} & I_{4 \times 4} \end{bmatrix}, \quad (10)$$

where $I_{4 \times 4}$ is the 4x4 identity matrix, $0_{4 \times 4}$ is the 4x4 null matrix and Δt is the frame sample time. After the particle propagation, the bounds check is applied, and the SURF algorithm is applied to each particle, obtaining their respective characteristic points.

G) Particle probabilities estimation: For determining the similarity of a particle with the reference particle, a pairing of the characteristic points is performed. In this pairing, the numbers of pairs of points whose metric is less than a threshold metric are counted. The relationship between the numbers of pairs of points with respect to the total number of characteristic points of the reference particle gives us the similarity probability.

H) Obtaining the most significant particles: To obtain the most significant particles, these are sorted according to their probability and the average value of the particles with a probability higher than the selected probability threshold, determine the position and size of the object to be followed.

G) Re-Sampling: To carry out the reassignment of the particles, the most probable particle obtained in the previous step is taken as the reference, and the new particle set is distributed according to the dispersion of each variable, always limiting them to the size of the screen. The adaptation of the dispersion of the variables according to the number of valid particles was implemented as a strategy of re-sampling, in such a way that the search zone is extended as the number of valid particles decreases.

6 Experimental results

Two experimental tests were carried out. The first one corresponds to track a six rotor "UAV" which was followed by a thermal camera (Tau 640 in the band of 8 to 14 microns), mounted on a positioning system. The second test corresponds to the same thermal camera mounted on the same vehicle taking a zenithal video where the target is marking a heliport. In both cases the object to be followed was not perfectly identified, in the first one part of the flight the UAV is behind trees of a grove, and in the second one, the marking of the heliport is affected by the infrared emissions of the terrain.

6.1. UAV tracking.

In a first experiment, the UAV flies hidden behind the trees to avoid detection, after which it starts a free flight. In both circumstances the particle filter is able to detect the UAV, even when it is almost completely hidden behind the trees. From the moment of the first detection, the particle filter remains locked throughout the remaining flight of the UAV.

In Fig. 8, part of the frame sequence shows the part of the flight where the UAV is first flying behind the trees, while the system begins distributing the particles in the largest possible area of the screen (Fig. 8 (a) and (b)).

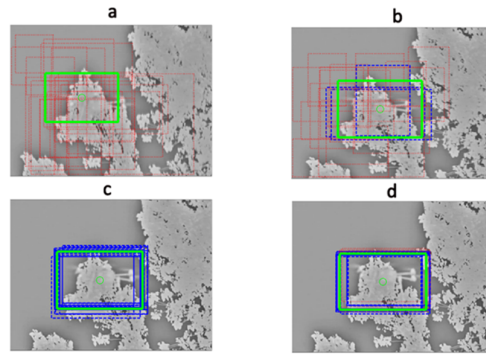


Fig. 8. Particles evolution (a)126.25 s, (b)126.30 s, (c)126.35 s, (d)126.40 s

In the frames of seconds, 126.35 and 126.40 (Fig 8 (c) and (d)), with the UAV still largely behind the trees, the particle filter is still able to detect it and “hook” it as soon as it partially appears. In this figure the particles are shown according to the selected probability threshold, in blue those above the threshold and in red those below, and in green the particle that the filter estimates as the best position and size of the object.

When the target leaves the trees, the filter remains locked during the whole flight, especially when the occlusion free flight is carried out, and the tracking is always tight (see Figs. 9, (a) to (d)).

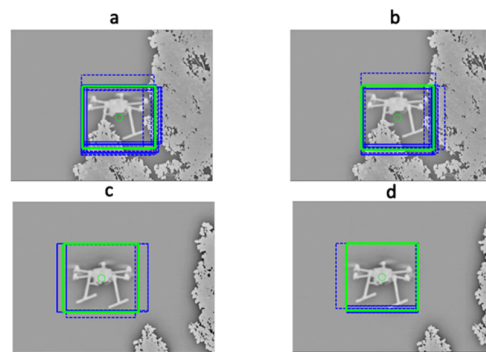


Fig. 9. During occlusion-free flight, subfigures (a) 127.30 s, (b) 127.35 s, (c) 129.15 s, and (d) 129.20 s.

In this case the UAV was also followed by a positioning system with the algorithm of detection by quantification. Therefore it is possible to establish a comparison between both algorithms (the quantification and the particle filter) through out of the absolute errors referred to the true position of the UAV. To do this the absolute errors (in pixels) of both algorithms were determined in the time frame between 126 and 129 seconds, just when the UAV leave the trees of a grove. In the case of the particle filter was running ten times, determining the absolute errors for each ones and the average value of the set.

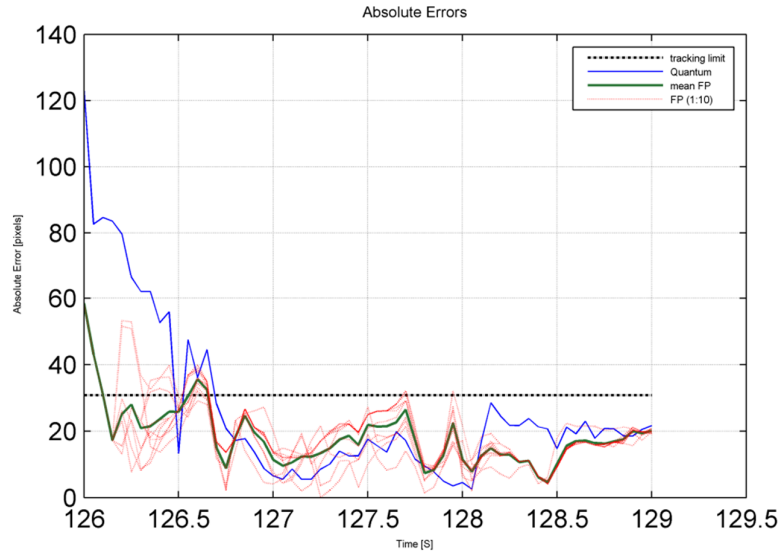


Fig. 10. Absolute errors in pixels, of the quantum and particle filter algorithm.

Fig. 10 shows the absolute errors of the 10 runs of the particulate filter and its average value, together with the absolute error of the quantization algorithm and the maximus radius in pixels from which the “hooked” UAV is considered. As can be seen in this figure, the response of the particle filter is much faster, entering in the “hook state” 0.5 seconds earlier and inclusive during the occlusion. After occlusion, (around second 126.8) the error stabilizes at a very low bound, and almost all generated particles are above the probability threshold.

6.2. Heliport tracking.

The second experiment was a flight over a heliport located in a park area, taken from zenithal videos. The take is interfered by the infrared radiation from the ground. In Fig. 11, the sequence of execution of the filter, selection of the objective (Fig. 11 (a)), initialization of the particles (Fig. 11 (b)), and locking during the flight (Fig. 11 (c) and (d)) can be observed.

Figure 12 shows the evolution of the particle filter, (which cannot be compared to other positioning reference). It can be observed that the number of valid particles indicates the fast match with the reference image, like in the previous experiment.

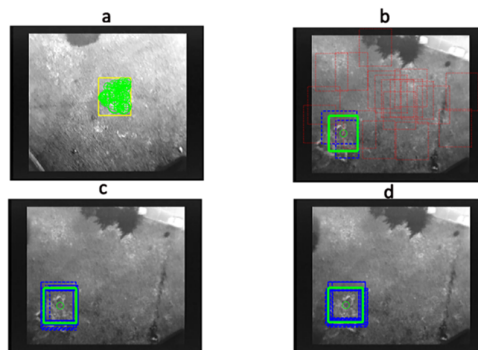


Fig. 11. Sequence of evolution, (a) Target, (b) 203.00 s, (c) 203.04 s and (d) 203.08 s.

7 Discussion and conclusions

The parameters used in this particle filter are NP (number of simultaneous particles used by the filter), PrTh (probability threshold to determine if a particle is considered similar to the reference), and SurfP (number of characteristic points of the SURF algorithm used for the determination of the similarity probability of a particle with respect to the reference). Based on these parameters, the following evaluations were made to determine the best set of parameters that make the filter have its best response. To compare the effect of these parameters on the filter response, we proceeded to analyse its temporal response as the number of valid particles, and the relative error between the positioning system and the particle filter.

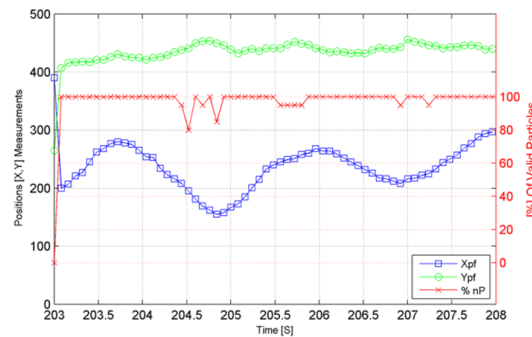


Fig. 12. Details of position of the particle filter, and particles above probability threshold.

Figure 13 shows the behaviour of the filter, as a function of the number of particles for values of 30, 50, 70 and 90. In this plot the dotted line corresponds to the relative errors, while the solid line corresponds to the number of valid particles. This last parameter is related to the temporal response, and the points in the lines represents the time of acquisition of the images, which sample time is 0.050 seconds. As can be seen, the number of particles does not affect the temporary response for the first lock. While, increasing the number of the filter particles it decreases its relative error, and presents greater stability to maintain the lock.

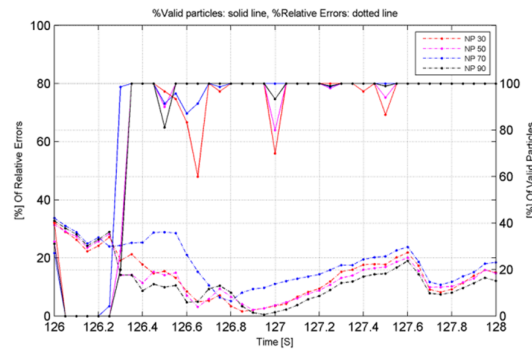


Fig. 13. Number of valid particles and relative error, for different values of particles.

Fig. 14 shows the behaviour of the filter, function of the number of characteristic SURF points for values of 30, 50, 70 and 90.

In this case the filter begins to have a good behaviour from a number of characteristic points of the SURF algorithm over 50. As this number increases, the detection improves when the target begins to leave behind the trees, and also improves the stability in keeping lock the target.

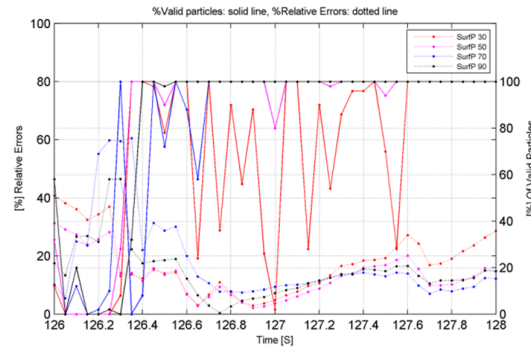


Fig. 14. Number of valid particles and relative error, for different values of SURF points.

The last parameter analysed is the similarity probability threshold that determine is a particle can be considered or not to obtain the target position. Fig 15 shows the filter behaviour for values of 86 %, 90 %, 94 % and 98 %.

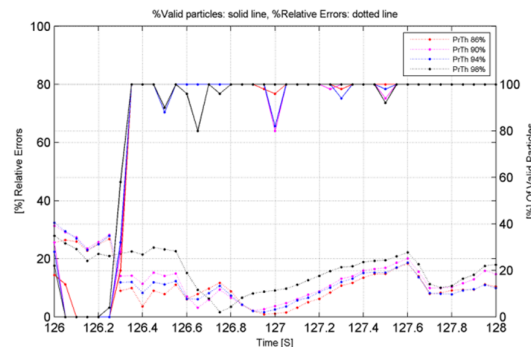


Fig. 15. Number of valid particles and relative error, for different values of probability threshold

The best response of the filter particles is for values equal to or higher than 94% in the similarity probability threshold. We can conclude that the minimum values in the initialization parameters that produce an acceptable response in the particle filter are: for the number of particles and the number of characteristic points of the SURF algorithm 70 or higher, and for the probability of similarity threshold 90% or higher. With values higher than these a fast response of the implemented algorithm of the particle filter to the locking of the reference object is observed, this responds in the time of two frames samples. As well as in the response to the tracking thereof, despite the non-optimal image or if the target is behind the trees when it beginning to leave them.

Another parameter that improved the particle filter response is the dispersion multiplying factor of the state variables. This multiplying factor is a function of the number of valid particles, such that, if the number of valid particles decrease, the multiplying factor is increased, covering a wider area of search. With this search strategy the filter maintains its stability in the response to the lock with the objective, in one frame. We can conclude that these preliminary results confirm the feasibility of unmanned night flights based on infrared video analysis.

As future work, some enhancements are proposed to achieve a better detection of the UAV when it is almost completely occluded by trees, or also in the case of the heliport targeting when the image is highly contaminated with background noise caused by atmospheric conditions. For this, our strategy consists of decomposing the reference image in N sub-images, with its corresponding identification. For each sub-image, the characteristic points are obtained together with their relative position in the original reference image. With this, we will have N particles filters each of one looking for a part of the reference image.

As a result, each particle filter gives us the most probable particle, together with the probability of similarity with the part of the object to be followed. After each filter obtains the most probable particle and

its probability of similarity, to determine the position of the object to be followed, we can sort in decreasing order the solutions of the particle filters according to the probability of similarity obtained. From there and starting with the first particle, we look for the particles that are located in the correct sequence. The average of these particles will give us the likely position of the target to follow. In case that no particles are found in the correct positions we can obtain the most probability position of the object to be followed as the position of the particle with the largest probability, or others means of integrating together the information provided by the N filters.

8 Acknowledgments

Special thanks to the staff of the Laboratories of the Institute of Scientific and Technical Research for Defence, dependent on the Technology Management, to the Thermal Imaging Laboratory of the Image Processing Department for providing the images of the experience, and to the Digital Techniques Laboratory of the Department of Applied Electronics for carrying out the flight with the unmanned aerial vehicle.

9 References

- [1] Amanda Berg, ‘Detection and Tracking in Thermal Infrared Imagery’, Linköping University, Sweden, 2016.
- [2] Santosh Bhusal, ‘Object Detection and Tracking in wide area Surveillance using Thermal Imagery’, Tribhuvan University, Nepal, 2015.
- [3] Zhang Shuang, Qin Yu-ping, ‘Mean-Shift Algorithm Apply for Infrared Imaging Tracking’, *Elsevier*, 2012.
- [4] Jiangtao Wang and Jingai Zhang, ‘Robust Object Tracking in Infrared Video via Adaptive Weighted Patches’, *Math. Comput. Appl.*, Dec. 2006.
- [5] Frederik S. Leira, et al., ‘Automatic detection, classification and tracking of objects in the ocean surface from UAVs using a thermal camera’, presented at the 2015 IEEE Aerospace Conference, 2015.
- [6] A. Hamdulla, G. Xiang, D. Tursun, ‘A Particle Filter and Fuzzy Clustering Based Algorithm for Tracking Dim Moving Multiple Point Targets in IR Image Sequence’, in *2009 WRI World Congress on Computer Science and Information Engineering*, 2009, vol. 7, pp. 205–209.
- [7] Hu Fanjun, Shi Wenjun, ‘Infrared Moving Multi-target Tracking Based on Particle Filter and FCM’, in *Conference on Computer Science and Electronics Engineering*, 2013, vol. 42.
- [8] F. Wang, E. Liu, J. Yang, S. Yu, and Y. Zhou, ‘Target tracking in infrared imagery using a novel particle filter’, *Chin Opt Lett*, vol. 7, no. 7, pp. 576–579, Jul. 2009.
- [9] Maybeck, Peter, *Stochastic models, estimation and control*, vol. 3. Academic Press, Inc, 1982.
- [10] Frank L. Lewis, Lihua Xie, and Dan Popa, *Optimal and Robust Estimation With an Introduction to Stochastic Control Theory*, Second. 2008.
- [11] Dr. Ciza Thomas, *Sensor Fusion and Its Applications*. National Institute of Technology (Norway), 2010.
- [12] David P. Landau, Kurt Binder, *A Guide to Monte Carlo Simulations in Statistical Physics*, Third Edition. Cambridge University Press, 2009.
- [13] Jari Kaipio, Erkki Somersalo, *Statistical and Computational Inverse Problems*, vol. 160. Springer Science, 2004.
- [14] Arnaud Doucet and Adam M. Johansen, ‘A Tutorial on Particle Filtering and Smoothing’, *Handb. Nonlinear Filter.*, vol. 12, Mar. 2012.

- [15] Fredrik Gustafsson, 'Particle Filter Theory and Practice with Positioning Applications', *IEEE Aerosp. Electron. Syst. Mag.*, vol. 25, no. 7, Jul. 2010.
- [16] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool, 'Speeded Up Robust Features', *Comput. Vis. Image Underst.*, vol. 110, no. 3, Jun. 2008.
- [17] H. Kandil and A. Atwan, 'A Comparative Study between SIFT- Particle and SURF-Particle', *Int. J. Signal Process. Pattern Recognit.*, vol. Vol. 5, no. No 3, Sep. 2012.
- [18] Roger A. Horn, Charles R. Johnson, *Matrix Analysis*, Second edition. Cambridge University Press, 2013.
- [19] Larry Venetsky, Ross Boczar, 'Optimization of Background Subtraction for Image Enhancement', *Mach. Intell. Bio-Inspired Comput.*, vol. 8751, Apr. 2013.
- [20] Alan M. McIvor, 'Background Subtraction Techniques', in *2004, IEEE International Conference on Systems*, 2004, vol. 4, pp. 3099–3104, vol4.
- [21] Dah-Chung Chang ; Wen-Rong Wu, 'Image contrast enhancement based on a local standard deviation model', in *1996, IEEE Nuclear Science Symposium. Conference Record*, 1996, vol. 3, pp. 1826–1830, vol3.
- [22] Prabhdeep Singh, Dr. A.K Garg, 'Non Uniform Background Removal using Morphology based Structuring Element for Particle Analysis', *Int. J. Comput. Appl.*, vol. 33, no. 6, Nov. 2011.