



**HAL**  
open science

# Moving Obstacles Detection and Camera Pointing for Mobile Robot Applications

Mustapha Lakrouf, Stanislas Larnier, Michel Devy, Nouara Achour

► **To cite this version:**

Mustapha Lakrouf, Stanislas Larnier, Michel Devy, Nouara Achour. Moving Obstacles Detection and Camera Pointing for Mobile Robot Applications. ICMRE 2017 2017 3rd International Conference on Mechatronics and Robotics Engineering , Feb 2017, Paris, France. 6p., 10.1145/3068796.3068816 . hal-01579420

**HAL Id: hal-01579420**

**<https://hal.science/hal-01579420v1>**

Submitted on 14 Sep 2017

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Moving obstacles detection and camera pointing for mobile robot applications

Mustapha Lakrouf

CNRS, LAAS, Université de Toulouse, 7 Avenue du Colonel Roche, Toulouse F-31400, France,  
LRPE laboratory, USTHB, BP32 El Alia, Bab Ezzouar Algiers, Algeria,  
Centre de Développement des Technologies Avancées (CDTA), cite 20 Aout 1956 Baba Hassan, Algiers, Algeria,  
mlakrouf@laas.fr

Michel Devy

CNRS, LAAS, Université de Toulouse  
7 Avenue du Colonel Roche  
Toulouse, France  
devy@laas.fr

Stanislas Larnier

AKKA Research  
6 impasse Michel Labrousse  
Toulouse, France  
stanislas.larnier@akka.eu

Nouara Achour

LRPE laboratory, USTHB  
BP32 El Alia, Bab Ezzouar  
Algiers, Algeria  
nouara.achour@usthb.dz

## ABSTRACT

Perception is one of the most important tasks for safe navigation of autonomous mobile robots in outdoor environments that are dynamic and unknown. This paper presents a real time system for detection and classification of dynamic objects using a 2D laser range finder (2D LRF) and a moving camera mounted on the mobile robot. An occupancy grid based detection is implemented using 2D LRF and moving objects are iteratively extracted from the global occupancy grid. A visual classification system is developed using HOG features and AdaBoost as learning process. Dynamic objects detection and visual classification results are coupled to point moving camera on the specified target (specified by mission supervisor) and to identify it. Finally, real time experiments results on the mobile robot in two different scenarios are presented.

## CCS CONCEPTS

•Computer systems organization → Robotics;

## KEYWORDS

Occupancy grid, moving objects detection, visual classification and pedestrian detection

## 1 INTRODUCTION

In our days, mobile robots are more and more used in different applications and navigate in complex environments. Outdoor environments are the most complex ones because of their dynamic and unknown aspects. To give to a mobile robot a height level of autonomy and safe navigation, a good and robust perception is required. To ensure that, a real time system for detection, classification and tracking of

mobile objects is necessary. Several approaches deal with the problem of Detection And Tracking Mobile Objects (DATMO). These approaches depend on the used sensors, chosen environment representation and mission objectives.

LRF sensors are ones of the most used sensors for moving objects detection [12], and vision is often used for targets classification. Some works [11, 13] present methods for detection, classification and tracking of moving objects using only 2D LRF. Other works treat jointly DATMO and SLAM problems (SLAMMOT). In [15], Vu corrects odometry localization by applying a scan matching method against OG which takes a counter of moving object by introducing a likelihood approach. Once a good location is obtained, updating incrementally the local OG allows to detect moving objects without a prior knowledge of the targets. In [2], and as a continuity of [15], authors add vision information, in different fusion levels, to classify targets into pedestrian or cars to improve tracking. Let us mention [8] where authors present an approach to map, detect, track and classify moving object in outdoor environment using a LRF sensor.

The aim of the work presented in this paper, which is part of the Air-Cobot project (introduced in Section 2), is to set up a perception system that gives to the mission supervisor informations (position) about mobile objects and to identify chosen target (is it human?, is he an operator?). Deployed sensors for dynamic targets detection are 2D LRF. To classify and identify targets, there are two possibilities for the moving cameras: Pan-Tilt-Zoom (PTZ) camera or firewire camera mounted on a Pan-Tilt Unit (PTU).

The localization is done by another work team on the same project. It is based on fusion of different localization methods using different sensors: odometry, Inertial Measurement Unit (IMU), relative localizations and GPS data.

Given this project and mission guidelines, our detection of moving objects is based on the global OG (limited work area: around a plane). For human visual detection, a HOG

based detector with AdaBoost learning process is used. A fusion step is done to point camera on chosen target and to classify it.

The rest of this paper is divided as follow. Section 2 give a presentation of the Air-Cobot project. In Section 3, we detail a detection system including its three parts: mobile object detection part based on OG using LRF sensor, vision classification part and data fusion to point moving camera on the chosen target and identify it. Experiments results depending on two different scenarios are given in Section 4. Conclusions and prospects are provided in Section 5.

## 2 AIR-COBOT PROJECT

### 2.1 Project presentation

Airplanes are inspected periodically during maintenance operations on an airport between flights or in a hangar for further inspection. The reduction in inspection time is a major objective for aircraft manufacturers and airlines. If maintenance operations are faster, this will optimize the availability of aircraft and reduce costs. Nowadays, the inspection is performed by human operators mainly visually, sometimes with some tools to evaluate defects. The French multi-partner Air-Cobot project aims to improve maintenance time and also traceability.



**Figure 1: The mobile platform Air-Cobot**

An autonomous and collaborative mobile robot is designed to help the aircraft inspection. It has sensors dedicated to inspection: PTZ camera and 3D scanner and other to navigation: firewire cameras, 2D LRFs, GPS and IMU. The chosen model for the 2D LRF is a Hokuyo UTM-30LX. Provided with wavelength of 870 nm lasers, it scans a semicircular 270 area. The robot has two on the front and rear, mounted on platines which can evolve into pan and tilt.

To navigate in the airport, the robot can go to an airplane parking thanks to geolocalization data, or by following its human operator [4]. To autonomously navigate around the airplane, the robot is able to use laser and vision methods to localize itself compared to the aircraft [6, 10]. Obstacle recognition and avoidance are also used in navigation mode [7]. The robot can inspect visually some items of the aircraft such as probes, static ports, trapdoors, latches and scan some fuselage parts [9, 14]. It has a tasks checklist to follow. The human operator controls the inspection diagnoses on its

tablet. He also checks visually the aircraft and can request additional non-destructive testing checks [4].

### 2.2 Use cases

This section presents four use cases where the moving obstacles detection and camera pointing for identification is useful and will improve the mission efficiency.

First use case is in navigation mode to reach the inspection site. The robot use GPS data to replay a previous learnt trajectory in the airport to reach an aircraft to inspect or going back to its docking station. In this state, all the camera systems can be used to point and recognize a moving obstacle.

Second use case is in navigation mode to reach each checking point. When navigating between checking points around the aircraft, the navigation cameras are used for pose estimation or visual servoing or visual odometry and the PTZ one is usually free.

Third use case is in inspection mode. When visually inspecting an item with the PTZ camera, the navigation cameras can be used for obstacle tasks if a moving obstacle arrived at proximity of the robot.

Finally, the fourth use case is in follower mode. When the cobot is in this mode, it has to fusion laser and video data to track and follow its human operator. Color image from the PTZ camera could be used in this case. The human operator will wear a distinctive color jacket.

## 3 DETECTION SYSTEM DESCRIPTION

### 3.1 Moving Objects Detection

Moving objects detection is done using only 2D LRF sensor. The robot environment is represented by an OG based on the laser data. Since the environment, in this work, is limited (around the plane) a global OG is constructed using localization given by another work team. In the following part, we will present how OG is constructed and moving objects extraction method.

**3.1.1 Occupancy Grid construction.** In OG representation, the robot environment is divided into a 2D grid of rectangular cells and each cell received a measure which has a real value in  $[0, 1]$  indicating the probability that the cell is occupied by an obstacle. A high value of occupancy grid indicates the cell is occupied and a low value means the cell is free. To construct a global OG, first we transform, at each time  $k$ , laser data or measurements  $z_k$  from local robot frame  $F_l$  to the global frame  $F_g$  (initial robot position) using transformation matrix  $T_k$  given by:

$$T_k = \begin{pmatrix} \cos(\theta r_k) & -\sin(\theta r_k) & x r_k \\ \sin(\theta r_k) & \cos(\theta r_k) & y r_k \\ 0 & 0 & 1 \end{pmatrix} \quad (1)$$

where  $(x r_k, y r_k, \theta r_k)$  represents the robot pose at time  $k$ .

Obtained measurements  $z_k|_{F_g}$  are used to determine free and occupied cells in the global grid at time  $k$  by applying Bresenham algorithm. Then the global OG is updated by

applying Bayesian update scheme [5] that provides a recursive formula under log-odds form:

$$\begin{aligned} & \log(O(M_i|x_{1:k}, z_{1:k})) \\ &= \log(O(M_i|x_{1:k-1}, z_{1:k-1})) + \log(O(M_i|x_k, z_k)) \\ & - \log(O(M_i)) \end{aligned} \quad (2)$$

where  $O(a|b) = \text{odds}(a|b) = P(a|b)/(1 - P(a|b))$ . The probability  $P(M_i)$  is the prior occupancy probability of the map which is set to 0.5 representing an unknown state, this makes this component disappears. The probability  $P(M_i|x_k, z_k)$  is called the inverse sensor model. It specifies the occupancy probability of a grid cell  $M_i$  based on a single sensor measurement  $z_k$  at location  $x_k$ .

**3.1.2 Moving objects extraction.** To extract moving objects from an OG, a Fast classification of static and dynamic environment is used [1]. At each iteration  $k$ , a global OG is constructed, as presented in previous section. To each cell  $M_i$ , two counters,  $\sigma_k(i)$  and  $\mu_k(i)$ , are associated in order to keep count of the number of times a cell has been observed free and the number of times it has been observed occupied respectively. It means, at time  $k$ , if the cell  $M_i$  is free,  $\sigma_k(i)$  is incremented, if it is occupied,  $\mu_k(i)$  is incremented and there is no incrementation otherwise. To give sense to the free and occupied counters, only the last  $J$  iterations are taken into consideration.  $J$  is defined experimentally.

To decide if the cell  $M_i$  is moving this counter is compared as follow:

$$\begin{cases} \text{moving} & \text{if } \sigma_k(i) > N \times \mu_k(i) \ \& \ OG_k(i) > 0.5 \\ \text{static} & \text{otherwise} \end{cases} \quad (3)$$

where  $N$  is an integer parameter defined experimentally that determine the sensibility to detect the change of cell state (from moving to static or the inverse).

To eliminate some false detection, moving cells which are adjacent to static objects are considered as static. Finally clustering of closest moving cells into moving objects is done using the Euclidian distance.

## 3.2 Visual Classification

The aim of this classification is to detect pedestrian using visual sensor. To construct this detection unit, a cascade of classifiers with a fixed size model based on HOG features is trained. It is then applied to each image in several scales and positions [3]. The selection of HOG features is obtained through a learning process using AdaBoost algorithm.

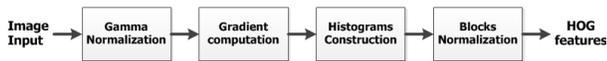


Figure 2: HOG features extraction chain

**3.2.1 HOG features extraction.** To extract HOG features, we begin by applying the Gamma normalization to the input image. In the second step, we calculate the gradient using first derivative filter as it is one of the most simple and

fast operators. The next step is histograms construction, where the image is divided into overlapping blocks. We use R-HOG descriptor with rectangular blocks of  $2 \times 2$  cells consisting of  $8 \times 8$  pixels. Overlapping blocks are obtained by horizontal and vertical sliding steps of 8 pixels. Finally, blocs are normalized. HOG features extraction chain is presented in Figure 2.

**3.2.2 Training process.** For training process, our images database is compounded of a selected part of INRIA Person Dataset [10] completed with our own pedestrian images. The output is a training set containing 1500 positive images and 2500 negative images.

Gentle-AdaBoost is used for training with cascade structure. The cascade is composed of  $k$  stages (strong classifiers) for which we precise a minimum hit rate  $d_i$  and a maximum false alarm  $f_i$ . The final detection rate  $D$  and the false alarm  $F$  are then given by:

$$D = \prod_{i=1}^k d_i \quad \text{and} \quad F = \prod_{i=1}^k f_i. \quad (4)$$

The detection cascade is of 28 stages where the minimum hit rate per stage is 0.999 and the maximum false alarm is 0.5. The final detection rate is then 0.97. Each stage is composed of weak classifiers, in our case HOG descriptors and 851 weak classifiers are used. A weak classifier have two possible output values  $v_1$  in case of rejection and  $v_2$  in case of positive decision.

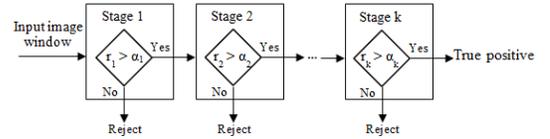


Figure 3: Classical cascade structure

In the classical cascade structure, described in Figure 3 all stages are of the same weight. A response  $r_k$  of each stage  $k$  given by:

$$r_k = \sum_{i=1}^l v_i \quad (5)$$

$v_i$  is the response of the weak classifier  $i$ . The response  $r$  is then compared to a threshold  $\alpha_k$ . This threshold represents the minimum response to a true positive window.

A weighted cascade structure is used in this work to speed up a classical cascade, a confidence factor  $C_l$  is calculated gradually at each stage  $l$  by:

$$C_l = \frac{1}{\omega_{max}} \sum_{i=1}^k \omega_i \quad (6)$$

Where  $\omega_{max}$  is the sum of positive decision values of weak classifiers all over the cascade stages. It is constant and calculated by Equation 7 and  $\omega_i$  is the weight of the response

$r_i$  defined as the sum of positive decision within stage  $i$  expressed by Equation 8.

$$\omega_{max} = \sum_{i=1}^k \sum_{j=0}^{l_i} v_{j+2} \quad (7)$$

$$\omega_i = \sum_{j=1}^{l_i} v_{j+2} \quad (8)$$

$k$  is the number of stages,  $l_i$  is the number of weak classifiers within stage  $i$  and  $v_{j+2}$  refers to the positive decision value of the weak classifier  $j$  within stage  $i$ . The weighted cascade structure is described in Figure 4.

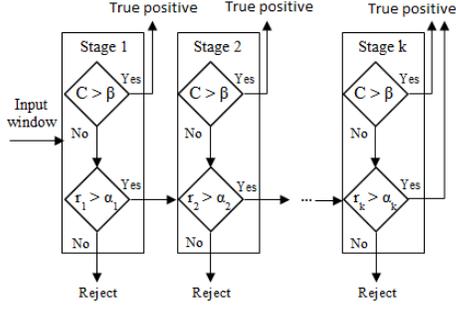


Figure 4: Weighted cascade structure

At the end of a stage evaluation, the confidence factor is compared to a threshold  $\beta$  defined experimentally to determine the minimum confidence value for a positive decision.

### 3.3 Keeping target in the camera field of view

After detecting moving objects by 2D LRF, our target will be the closest moving objects to the robot (using Euclidian distance). The visual classification is applied to determine if our target is a pedestrian or not. But, the target must be seen by the camera and for it we use a moving camera to point the target and keep it as long as possible in the camera Field Of View ( $FOV$ ). The flow chart presented in Figure 5 summarize our system architecture.

*Assumption (1)*: given that the object to be detected by vision is the pedestrian, it have a contact point with a ground, and our working ground is flat, so we can set tilt camera to horizontal position knowing that it is mounted on the robot such as that is 1 m elevated from ground.

**3.3.1 Pointing target:** Under the Assumption (1) and to point camera to a given target we must only calculate the corresponding pan position. To better understanding this, Figure 6 gives aerial view of the robot and target with different frames and angles. Let  $(Tx_k, Ty_k)$  the target position at time  $k$  in the absolute frame,  $(xr_k, yr_k, \theta r_k)$  the robot pose at time  $k$  and  $\phi_k$  pan position of the camera at time  $k$ . The target position  $(tx_k, ty_k)$  in the robot frame is

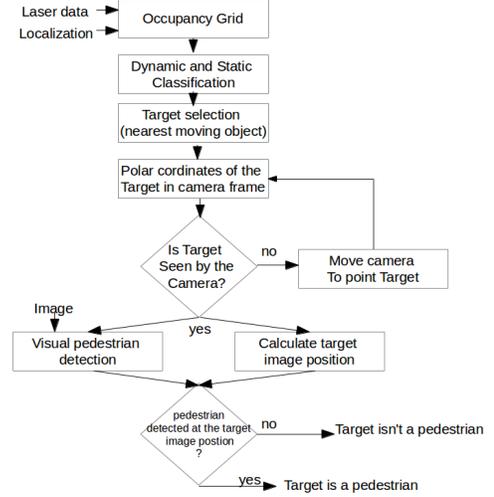


Figure 5: System architecture

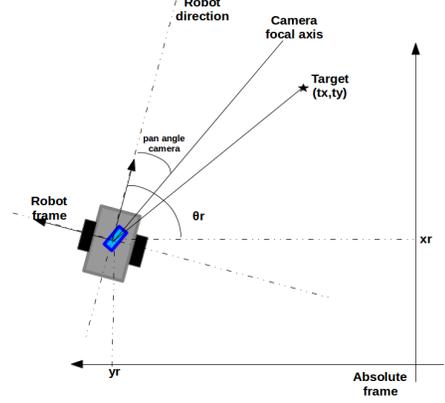


Figure 6: Absolute, robot and camera frames

calculated by applying the following transformation matrix  $T_{f,k}$ :

$$T_{f,k} = \begin{pmatrix} \cos(\theta r_k) & \sin(\theta r_k) & -xr_k \\ -\sin(\theta r_k) & \cos(\theta r_k) & -yr_k \\ 0 & 0 & 1 \end{pmatrix} \quad (9)$$

then the target position  $(tx_k, ty_k)$  is transformed from the Cartesian coordinates to the Polar coordinates  $(r_k, \alpha_k)$ :

$$(r_k, \alpha_k) = \left( \sqrt{tx_k^2 + ty_k^2}, \arctan\left(\frac{ty_k}{tx_k}\right) \right) \quad (10)$$

and  $\alpha_k^{cam\ frame} = \alpha_k - \phi_k$  represent  $\alpha_k$  in the camera frame.

If target is out of field of view  $(-FOV/2 < \alpha_k^{cam\ frame} < FOV/2)$  we move camera to  $\alpha_k^{cam\ frame}$  pan position, else (if target is seen) we calculate target position in image.

**3.3.2 Target position in image:** Given the 2D target position  $(Tx_k, Ty_k)$  at time  $k$  and Assumption (1) still required, we can calculate horizontal position  $Tu_k$  of the

target in the image at time  $k$  by defining a linear function  $f$  such that:

$$Tu_k = f(\alpha_k^{cam\_frame}) = A * \alpha_k^{cam\_frame} + B \quad (11)$$

Knowing that the  $FOV$  and horizontal resolution  $W$  of the camera, then  $A$  and  $B$  are given by:

$$A = W/FOV \text{ and } B = W/2 \quad (12)$$

Finally, and to do fusion between laser based target detection and visual classification, we can say that our target is a pedestrian if we detect by vision a pedestrian at the image position  $(P_u, P_v)$  that verify:  $|P_u - Tu_k| < Th_u$  with  $Th_u$  is a threshold to be defined experimentally.

## 4 EXPERIMENTS

Different experiments were performed to evaluate the developed system. This work is implemented on the 4-wheel skid-steering mobile robot, presented in Figure 1. An Hokuyo sensor (2D LRF) is used for moving objects detection and two moving cameras (PTZ camera and stereo cameras mounted on the Pan Tilt Unit) for pointing target and pedestrian detection. The software architecture of the robot is ROS based. Each elementary unit of our system is implemented in a separately node. This modular and parallel architecture gives a flexibility and portability to our system and reduces the execution time. Our detection system presents a real time processing performance, and all experiments are done online. The laser based detection part can run at frequency of 30 hz with map of size 60 m  $\times$  80 m and 0.01 m cell size. The pedestrian detection runs at the frequency of 25 hz with images of size 572  $\times$  434. The whole system can run at frequency of 25 hz.

To evaluate the system proposed in this paper, two experiments are presented in this section.

### 4.1 Experiment with static robot

The robot in the first experiment is static. Two pedestrians are moving in the robot environment. They must move such that the selected target (nearest moving object) switches between them. The left image in Figure 7 provides the workspace view of this experiment.

Figure 8 provides some results. The rows represent the experiment time evolution. The upper lign of images represents the OG of the environment at the same time that the corresponding snapshots of the bottom lign. Dynamic objects are represented by black cells in OG. The camera images contains pedestrian detection (white or red boxes) and target position (two vertical red line) and the red box represents that the target is detected as the pedestrian. In these results, one can note, firstly, that the detection of the moving objects by laser sensor using OG is acceptable. Secondly, the target (nearest moving object) is identified by vision detection. The camera is moved to point in direction of the target as it is illustrated in the columns of Figure 8.



Figure 7: Workspace views of the two experiments

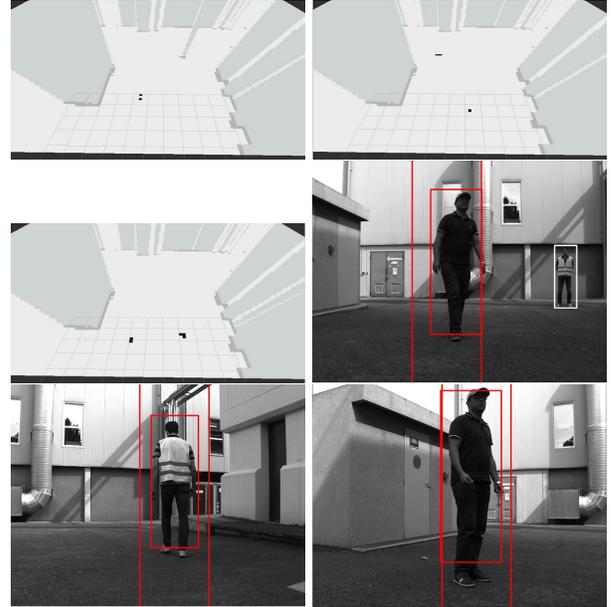
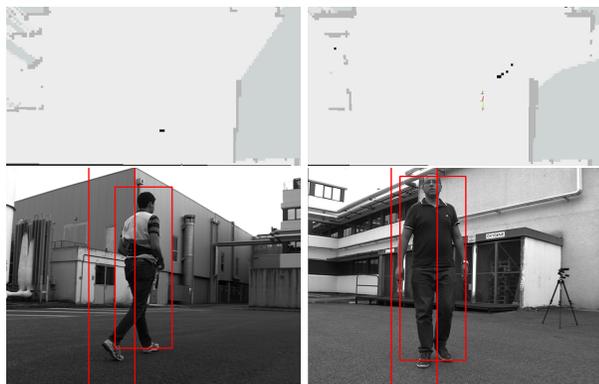


Figure 8: Static robot: moving objects detection using OG (first row images) and target identification (second row images); colons represent experiment time evolution.

### 4.2 Experiment with moving robot

The second experiment is similar to the first except that the robot and the pedestrians are moving. The right image in Figure 7 provides the workspace view of this experiment. Figure 9 provides some results. The pictures and legend conventions are the same. Obtained results show that the moving object detection give a good results when robot is moving and the moving camera keep the target into field of view then target is identified.



**Figure 9: Moving robot: moving objects detection using OG (first row) and target identification (second row); colons represent experiment time evolution.**

## 5 CONCLUSIONS AND PROSPECTS

In this paper, a real time multi sensor system for detection and classification of moving objects is presented. An OG based approach is used to represent robot environment and to extract moving objects by fast classification algorithm. In the second part of this work, a vision based pedestrian detection is applied to identify target which is kept in the field of view by pointing the moving camera. The system detects and identifies the moving target if it is a pedestrian.

The method proposes a solution to a real industrial problem. The obtained results illustrates that determining the target position in the image and pointing cameras improve our target identification respecting real time constraints.

In the future, an addition to our system will be a multi target tracker to predict pointing position of a target and to track it when it is occluded. After being tested in the presented outside environment, the proposed approach will be validated for the Air-Cobot project thanks to a real context with aircrafts with human operations around it.

## ACKNOWLEDGMENT

This work is part of the Air-Cobot project (<http://aircobot.akka.eu>) approved by the Aerospace Valley world competitiveness cluster. We gratefully acknowledge financial support from the French Government via the Single Inter-Ministry Fund (FUI). The other partners of the Air-Cobot project (Airbus Group, Armines, 2MoRO Solutions, M3 Systems and Strla) are acknowledged for their support.

## REFERENCES

- [1] Q. Baig, M. Perrollaz, J. Botelho, and C. Laugier. 2012. Fast classification of static and dynamic environment for Bayesian Occupancy Filter (BOF). In *IROS12 4th Workshop on Planning, Perception and Navigation for Intelligent Vehicles*.
- [2] R.O. Chavez-Garcia and O. Aycard. 2015. Multiple Sensor Fusion and Classification for Moving Object Detection and Tracking.

- [3] A. Chayeb, N. Ouadah, Z. Tobal, M. Lakrouf, and O. Azouaoui. 2014. HOG based multi-object detection for urban navigation. In *Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on*. IEEE, 2962–2967.
- [4] F. Donadio, J. Frejaville, S. Larnier, and S. Vetault. 2016. Human-robot collaboration to perform aircraft inspection in working environment. In *Proceedings of 5th International conference on Machine Control and Guidance (MCG)*.
- [5] D. Fox, W. Burgard, and S. Thrun. 2005. *Probabilistic Robotics. Intelligent Robotics and Autonomous Agents*. The MIT Press.
- [6] J. Frejaville, S. Larnier, and S. Vetault. 2016. Localisation partir de donnees laser d'un robot naviguant autour d'un avion. In *Reconnaissance des Formes et Intelligence Artificielle (RFIA 2016)*.
- [7] M. Futterlieb, V. Cadenat, and T. Sentenac. 2014. A navigational framework combining Visual Servoing and spiral obstacle avoidance techniques. In *Informatics in Control, Automation and Robotics (ICINCO), 2014 11th International Conference on*, Vol. 02. 57–64.
- [8] G. Gate and F. Nashashibi. 2009. An approach for robust mapping, detection, tracking and classification in dynamic environments. In *Advanced Robotics, 2009. ICAR 2009. International Conference on*. IEEE, 1–6.
- [9] I. Jovancevic, S. Larnier, J.-J. Orteu, and T. Sentenac. 2015. Automated exterior inspection of an aircraft with a pan-tilt-zoom camera mounted on a mobile robot. *Journal of Electronic Imaging* 24, 6 (november 2015).
- [10] I. Jovancevic, I. Viana, J.-J. Orteu, T. Sentenac, and S. Larnier. 2016. Matching CAD Model and Image Features for Robot Navigation and Inspection of an Aircraft. In *Proceedings of the 5th International Conference on Pattern Recognition Applications and Methods*. 359–366.
- [11] A. Mendes, L.C. Bento, and U. Nunes. 2004. Multi-target detection and tracking with a laser scanner. In *Intelligent Vehicles Symposium, 2004 IEEE*. IEEE, 796–801.
- [12] C. Mertz, L.E. Navarro-Serment, R. MacLachlan, P. Rybski, A. Steinfeld, A. Suppe, C. Urmson, N. Vandapel, M. Hebert, C. Thorpe, and others. 2013. Moving object detection with laser scanners. *Journal of Field Robotics* 30, 1 (2013), 17–43.
- [13] K. Rebai, A. Benabderrahmane, O. Azouaoui, and N. Ouadah. 2009. Moving obstacles detection and tracking with laser range finder. In *Advanced Robotics, 2009. ICAR 2009. International Conference on*. IEEE, 1–6.
- [14] T. Villemot, S. Larnier, and S. Vetault. 2016. Dtection d'amers visuels pour la navigation d'un robot autonome autour d'un avion et son inspection. In *Reconnaissance des Formes et Intelligence Artificielle (RFIA 2016)*.
- [15] T.-D. Vu, J. Burtle, and O. Aycard. 2008. Grid-based localization and online mapping with moving objects detection and tracking: new results. In *Intelligent Vehicles Symposium, 2008 IEEE*. IEEE, 684–689.