Computers in Industry 86 (2017) 26–33

Contents lists available at ScienceDirect

Computers in Industry

journal homepage: www.elsevier.com/locate/compind

# Visual recognition of aircraft mechanical parts for smart maintenance

Claudio Cusano [a,1], Paolo Napoletano [b,1,*]

[a] Department of Electrical, Computer and Biomedical Engineering, Università degli Studi di Pavia, Via Ferrata 1, 2700 Pavia, Italy
[b] Department of Informatics, Systems and Communication, Università degli Studi di Milano-Bicocca, Viale Sarca 336, 20126 Milano, Italy

ARTICLE INFO

*Article history:*
Received 29 June 2016
Received in revised form 7 November 2016
Accepted 4 January 2017
Available online

*Keywords:*
Aircraft maintenance
Computer vision
Local descriptors
Object detection

ABSTRACT

Recent studies experienced the use of advanced tools for smart aircraft maintenance and inspection. These tools often require the use of computer-vision based technologies to recognize and track a given aircraft mechanical part in order to make it possible to show additional information to a technician on a suitable display. In this paper we propose a visual recognition module of aircraft mechanical parts that has been included in a prototype system designed for the smart maintenance of the Alenia-Aermacchi M346. The evaluation, carried out on real aircrafts, considers different kind of maintenance operations that require the recognition of 20 different mechanical parts. The visual recognition module has been tested under different imaging conditions and varying the scale and the orientation of the parts of interest. The results confirm the feasibility of our proposal also in such a very challenging and realistic condition.



## 1. Introduction

Maintenance and inspection operations of aircraft structures have a great impact on the life cycle cost of an aircraft [1–3]. During the periodic inspection and maintenance operations, technicians follow an approved baseline program specially designed for each aircraft. These operations includes the retrieval of information from suitable technical documentation, verification and reporting activities [4]. The use of information and communication technologies, as well as artificial intelligence techniques, can help to make these operations more cost effective with the same or better reliability with respect to traditional approaches [5–8].

In recent years, some researchers explored the use of augmented reality in aircraft maintenance and inspection as an effective way for displaying additional information about objects of interest [9–12]. In particular, De Crescenzio et al. [13] developed a prototype system for aircraft maintenance training and operations support. The system includes: a head-mounted display to show augmented reality; a marker-less camera pose estimation to track mechanical parts by using computer-vision based techniques; an efficient authoring procedure that enables quick and flexible creation of digital content. The tracking of mechanical parts has been done by using the SURF (Speeded-Up Robust Features) local invariant features. The prototype validation included 10 people that repeated the oil check procedure on three different aircrafts.

Jo et al. [14] proposed a system to simplify the aircraft technicians maintenance tasks, minimize operation errors and time-related costs. The system is composed mainly of three modules: the augmented reality module that includes a module for the tracking of mechanical parts; the knowledge-based system module to handle the technical documents; a graphical interface made of small windows and tab controls. The vision-based tracking, annotation, and recognition is done by using SURF features and self-similarity image matching.

De Crescenzio et al. [13] and Jo et al. [14] argued mainly on the feasibility of augmented reality for aircraft maintenance operations from the user usability point of view. None of the existing studies reported on the robustness of the computer vision techniques in recognizing and tracking mechanical parts. In both papers, authors implicitly validated the goodness of the

**Fig. 1.** Alenia-Aermacchi M346.

* Corresponding author.
*E-mail addresses:* claudio.cusano@unipv.it (C. Cusano), paolo.napoletano@disco.unimib.it (P. Napoletano).

[1] These authors contributed equally to this work.

http://dx.doi.org/10.1016/j.compind.2017.01.001

**Fig. 2.** Examples of some panels of the aircraft. Each panel contains several internal parts.

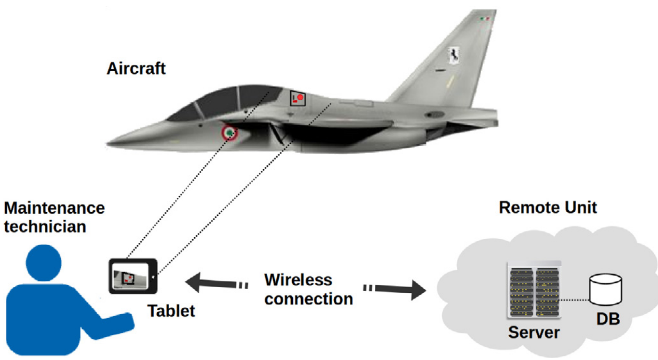


**Fig. 3.** Overview of the architecture of the smart maintenance system.

underlining computer-vision-based technologies by evaluating the prototypes efficiency and usability. In addition, both papers reported only on one maintenance operation: De Crescenzio et al. [13] experimented only on the oil check procedure while Jo et al. [14] experimented only on the maintenance task of the Airbus landing gear. These on-site experimentations required the recognition of only one mechanical part. One limitation of previous studies is the lack of a wide and convincing on-site experimentation that includes different imaging conditions and computer vision recognition metrics.

This paper proposes a robust method for recognition of aircraft panel and internal mechanical parts based on computer vision techniques. This method has been included in a smart aircraft maintenance system that has been developed within the "Smart Maintenance" Project.[2] The main scope of the project is to help aerospace companies, manufacturers, operators and service providers to reduce the costs of direct job needed for maintenance procedures [15]. Fig. 3 shows the architecture of the smart aircraft maintenance system developed within the project. The system is designed for the maintenance of the Alenia Aermacchi[3] M346 aircraft represented in Fig. 1. The maintenance procedure includes the inspection of panels and internal mechanical parts (see some examples reported in Fig. 2).

The maintenance system includes a technician, an aircraft, a remote server, an off-the-shelf tablet device and a mobile application (named "smart maintenance") developed for the project. The technician uses the application running on the tablet to take pictures of the panel of interest for a given maintenance operation. The application sends the picture to a web service, hosted by the server, that processes it by using computer vision techniques. The result consists in the identification of the panel and of the mechanical parts portrayed in the picture.

The computer vision module works as follow: given an image, it detects the presence of panels and of parts of interest by comparing the image itself with similar images that have been previously annotated by manually drawing the polygonal contours of the elements to recognize. The experiments considered 248 images portraying eight different panels and 20 different mechanical parts of the Alenia-Aermacchi M346. The panels and parts have been selected by experts from Alenia-Aermacchi because they are representative of different kind of maintenance operations. Each panel and each part have been acquired under different imaging conditions simulating the normal operating conditions encountered by the technicians performing the maintenance operations. Therefore, the appearance of the mechanical parts in the pictures are characterized by a large degree of variability in terms of lighting conditions, scale and orientation as depicted in Fig. 4.

The information about the panel and the mechanical parts recognized by the system could be exploited in several ways. For instance, it may be used to keep a visual log of the operations, to assist the training of the personnel, to provide additional technical information, etc. In this work we focus on the use case in which the maintainer needs to access to the technical documentation (drawings, 3D models, checklists, operational instructions, etc.) associated to one ore more mechanical parts involved in a given maintenance operation. To do so he takes a picture of the panel hosting the parts he is interested in, and then waits for the response of the server. The server will send back the same picture with the recognized parts highlighted by a colored boundary. Then, the maintainer selects one of the boundaries to query the remote database for the information related to the corresponding part. After he finished to read the documentation, he can proceed by selecting a different part, or by taking a new picture. The complete use case, that is summarized in Fig. 5, has been implemented in an experimental prototype which includes the mobile application, the remote service, and the remote database.

The paper is organized as follows: Section 2 presents an overview of methods for object recognition. Section 3 describes the proposed approach for the visual recognition of mechanical panels and parts of an aircraft. Section 4 describes the experimental protocol and the results obtained. Finally, Section 5 summarizes the work and highlights some promising directions for future research on this topic.

## 2. Methods for detection of mechanical parts

Object detection consists in determining whether or not a given object (or an instance of a given category of objects) is represented in an image and, in the positive case, where it is. Usually the object is defined by one or more reference images (called *templates*) where its location has been manually specified, for instance by providing a polygonal outline of its boundaries. In the literature, object detection has been approached with two main approaches [16]:

- *template-based* methods, in which the pixels of the templates are directly compared with the pixels in the input image;
- methods based on *local descriptors*, where salient points in the input image are selected and mathematically described; the description of each point is then compared with similar descriptions computed for the salient points in the templates.

[2] http://www.smartmaintenance.it.
[3] http://www.leonardocompany.com/.


**Fig. 4.** Some examples for each of the 20 mechanical parts included in the experimentation. Each example is a crop from a larger picture, and has been resampled to a uniform width.

We decided to focus on the second approach because it is more robust with respect to viewpoint variations, a property that is very useful when there is the need to introduce as few constraints as possible in the image acquisition process. In the simplest scenario, the input is a pair of images: one in which an element of interest has been previously outlined (i.e. the template), and one (that we will call the *input* image) in which we want to determine whether or not there is another instance of the same element (in that case we also want to identify its boundary in the image plane). The main steps of the detection procedure are:

1. detection of salient points (also called *keypoints*) in the two images;
2. computation of the *descriptor* associated to each keypoint;
3. matching between pairs of keypoints from the two images on the basis of the similarity between their descriptors;
4. search for a geometric relationship that is able to "explain" the spatial layout of the matching pairs of keypoints.

Fig. 6 illustrates the four main steps, that are described in greater detail in the following sections.

### 2.1. Detection of the keypoints

The detection process is based on the matching between pairs of keypoints. It is very important that their location is chosen in such a way that the region surrounding them can be recognized with sufficient reliability even when the images are taken from a different viewpoint, under different lighting conditions etc. Moreover, there must be a high chance that the similarity between two keypoints is not accidental, that is, their similarity must be indicative of the fact that they correspond to identical points in the acquired scene. To achieve this, keypoints must be chosen in the

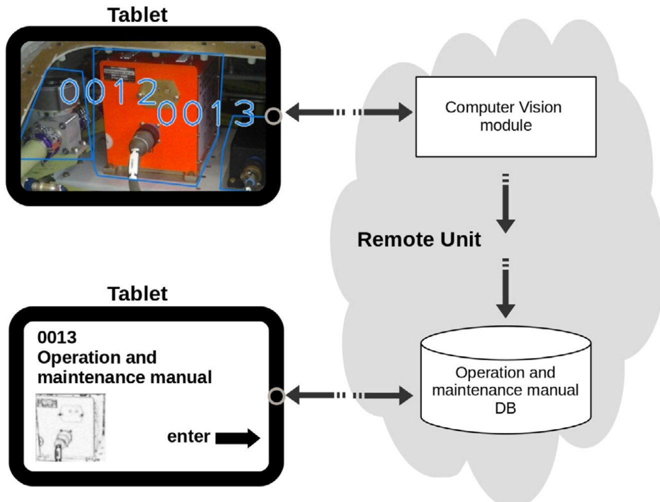


**Fig. 5.** Schematic view of the use case considered in this work: (i) the operator takes a picture; (ii) the computer vision module recognizes the parts of the aircraft; (iii) the operator selects one of the parts; (iv) the corresponding operation and maintenance manual is retrieved from the remote database; (v) the information is used to carry out with maintenance operation.

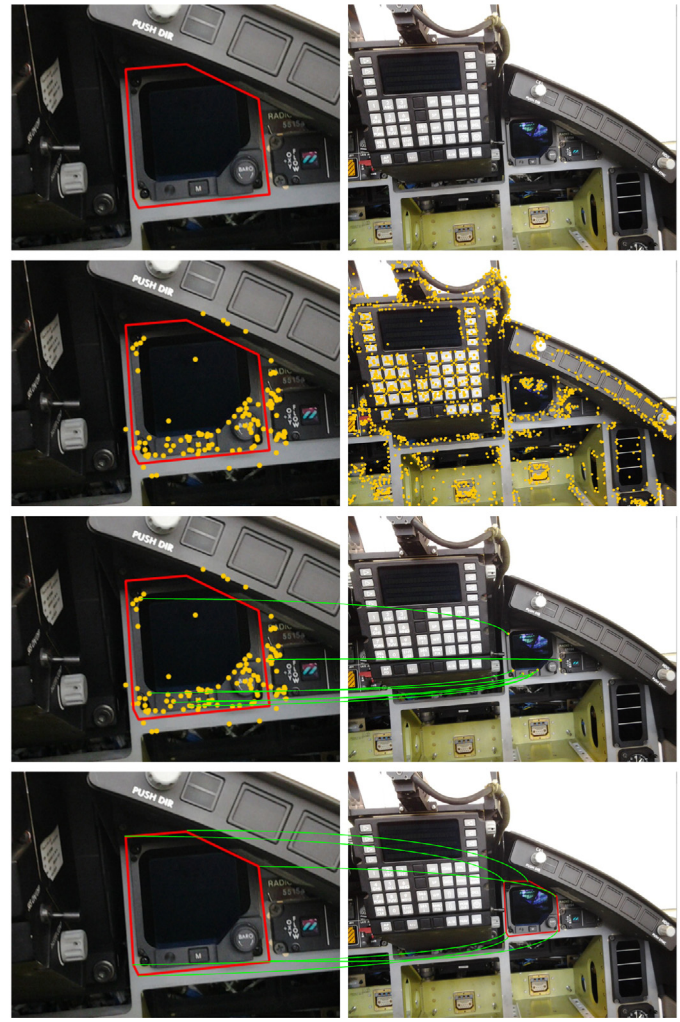

**Fig. 6.** The main steps of the recognition procedure in which the outlined part in a template image (left column) is searched in an input image (right column). From top to bottom: template and input images; keypoint detection; matching between the descriptors in the two images; boundary of the recognized part, obtained after the estimation of the geometric transformation relating the matching keypoints.

proximity of strong discontinuities along multiple directions (for instance, corners of objects). To do so, several operators have been proposed in the past. Among the others, we can mention the Harris corner detector [17] and its improved Harris-Laplace version [18], the Differences of Gaussian (DoG) operator [19], the Speeded Up Robust Features (SURF) [20]) and the Features from Accelerated Segment Test (FAST) [21]. In particular, the last three (DoG, SURF and FAST) have the advantage of being invariant with respect to rotation and scale, and, according to the comparison made by Tuytelaars and Mikolajczyk [22], they are characterized by having good degrees of repeatability (low sensitivity to the viewing conditions), accuracy (precision of localization), and robustness (low sensitivity to image noise, discretization effects, compression artifacts, blur, etc.). Among these SURF and FAST have been designed to optimize the speed of computation, sometimes at the expense of small decrease in accuracy with respect to DoG.

For the reasons outlined above, and considering that in our application all the computation is performed on the server, we chose to use the DoG operator. The operator consists in performing two convolutions of the images with two Gaussian filters differing in the scale parameter

$$D = (I \otimes G(k\sigma)) - (I \otimes G(\sigma)), \tag{1}$$

where $I \otimes G(\sigma)$ represents the convolution between the input image $I$ (converted to gray scale) and a Gaussian filter with parameter $\sigma$, and where $k > 1$ is a constant defining the relative difference between the scales. The absolute value $|D|$ captures the amount of discontinuity of the points in the image at the given scale. The operation is repeated for multiple scales, and the local maxima in the image plane and across scales, are selected as candidate keypoints. An orientation is assigned to the candidates by taking the direction of maximal local variation at their locations. Finally, candidates of ambiguous position, scale, or orientation, are discarded.

### 2.2. Computation of the descriptors

For each detected keypoint a numerical description is computed to be used for its matching. These descriptors are called "local" because they represent a neighborhood of the keypoint. They are computed by taking into account the scale and orientation of the keypoints. A multitude of local descriptors have been proposed in the literature [23], including the Scale Invariant Feature Transform (SIFT) [24], the Speeded Up Robust Features (SURF) [20], the Binary Robust Independent Elementary Features (BRIEF) [25], the Binary Robust Invariant scalable keypoints (BRISK) [26], and the Oriented FAST and rotated BRIEF (ORB) [27]. Among these we choose to adopt SIFT since, according to several comparative studies [23,28], they often outperform the others that, in fact, have been mainly designed to approximate the accuracy of SIFT but with a fraction of the computational cost.

SIFT descriptors are computed from a square region around the keypoints. The size and the rotation of the region depends on the scale and orientation of the keypoint. The square is divided in $4 \times 4$ tiles and for each tile a eight-bin histogram of the direction of the gradient of the image is computed. The final 128-dimensional descriptor is obtained by concatenating the 16 eight-dimensional histograms. To avoid local high contrast measurements from being given too excessive emphasis, Lowe proposed a two-stage normalization, where the entries after a first-stage unit sum normalization are limited to not exceed the value of 0.2, and then the modified image descriptor is normalized to unit sum again.

### 2.3. Pairwise matching

The recognition of panels and parts relies on the detection of corresponding points in the template and in the input images. A simple approach for detecting these correspondences consists in the exhaustive comparison of all possible pairs of keypoints. The comparison is based on the computation of a distance function between the descriptors. For instance, the Euclidean distance is often used to compare SIFT descriptors. Each keypoint in the input image is compared with all the keypoints in the template and the most similar one (i.e. that with the closest descriptor) is selected. If the distance between the descriptors is below a set threshold then the pair of keypoints is considered as a valid correspondence.

The quality of the correspondences greatly influences the reliability of the final recognition. Therefore, it is common to include additional criteria to reject dubious correspondences. A widely used one is to reject those case where a keypoint in the input image shows a high similarity with respect to multiple keypoints in the template image. More precisely, to accept a correspondence involving a given keypoint in the input image it is required that the ratio between the distances with the two most similar keypoints in the template image is above a set threshold.

The exhaustive search can be too expensive when there are many keypoints in the images. A strategy for a more effective selection of the correspondences has been proposed by Muja and Lowe [29]. Their approach, called FLANN (Fast Approximate Nearest Neighbors), consists in an approximate search where the closest keypoints is not guaranteed to be found. The loss in accuracy is balanced by a huge speed-up. The approximate search is justified by the fact that no exact method is known that is faster than the exhaustive search. FLANN combines two methods for the approximate search: the use of randomized *kd*-trees [30] and that of hierarchical *k*-means clustering [31]. The choice between the two methods, and the configuration of their parameters is made in two steps: first the parameters space is sampled and is chosen the combination minimizing the cost function

$$cost = \frac{s + w_b b}{(s + w_b b)_{opt}} + w_m m, \tag{2}$$

where $s$ denotes the time required to search all the keypoints, $b$ is the time required to initialize the algorithm and $m$ is the ratio between the amount of memory needed to initialized the algorithm and that required to store all the keypoints. The weights $w_b$ and $w_m$ determine a trade-off between computation time and memory usage. In a second step the Nelder-Mead optimization method is used to refine the values previously found.

### 2.4. Estimate of the geometric transformation

Not all the correspondences are reliable. Some of them may represent pairs of similar but distinct details in the images. From the set of all the correspondences found only those that define a coherent geometric transformation must be taken into account. The basic assumption is that if there exist a geometric transformation capable of explaining a large fraction of correspondences, then it is possible to conclude that the object in the template does actually occur in the input image. The transformation itself could then be used to infer the location of the object found. Since the object we are searching for could have been subject to rotations and translations in the 3D space, and since its image is the result of a projective transformation from the 3D space to the image plane, then the geometric transformation to be determined must be a homography.

In projective geometry homographies are linear transformations and thus they can be represented by a $3 \times 3$ matrix $H$ of coefficients:

$$s_i \begin{bmatrix} x'_i \\ y'_i \\ 1 \end{bmatrix} \simeq H \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix}, \tag{3}$$

where $x_i$, $y_i$ are the coordinates of the $i$-th keypoint in the input image, $x'_i$, $y'_i$ are the coordinates of the corresponding keypoint in the template image. In projective geometry such a relationship can be determined up to the scaling factor $s_i$. Moreover, the relationship is approximate, due to several factors including the acquisition noise, the digitization process of the images and the inaccuracies in the localization of the keypoints. The inexactness of the relationship suggests to use a least squares fitting approach to find $H$. More precisely, it is common to find $H$ by minimizing the *reprojection error* [32]:

$$\sum_i \left( x'_i - \frac{h_{11}x_i + h_{12}y_i + h_{13}}{h_{31}x_i + h_{32}y_i + h_{33}} \right)^2 + \left( y'_i - \frac{h_{21}x_i + h_{22}y_i + h_{23}}{h_{31}x_i + h_{32}y_i + h_{33}} \right)^2. \tag{4}$$

This formulation of the problem assumes that no false correspondences exist. The presence of false correspondences (outliers) can introduce large distortions in the reprojection errors. An optimization strategy that is robust with respect to the presence of outliers is therefore required. A method with this property is RANSAC (RANdomized SAmple Consensus) [33].

The RANSAC method works by iteratively selecting a random subset of correspondences. The selected correspondences are considered as possible inliers (i.e. non-outliers), and this hypothesis is verified by optimizing $H$ on them by minimizing the reprojection error. The resulting matrix $H$ is evaluated by counting how many correspondences (inliers and outliers) agrees with the geometric transformation that it defines (i.e. how many of them have an individual reprojection error below a set threshold). If this number is large enough then $H$ is refined by taking into account all the concordant correspondences. This procedure is repeated a set number of times, producing each time a matrix that is rejected due to the low number of concordant correspondences, or a "correct" matrix together with its reprojection error on the concordant correspondences. Finally, the matrix with the lowest error is selected. If all the generated matrices are rejected, then a failure in the search is declared.

Several variants of RANSAC have been proposed so far, differing in the criterion used to detect the outliers (MSAC [34]), in the optimization step (LO-RANSAC [35]), in the use of *a-priori* knowledge about the data (PROSAC [36]) etc. According to the work of Choi et al. these variants can outperform standard RANSAC in the context of homography estimation [37]. However, in this work we adopted the standard version and we deferred the experimentation of the alternatives to our future investigation.

## 3. Recognition of aircraft mechanical parts

The methods described in the previous section have been implemented and used to build a system for the automatic detection of panels and mechanical parts. More in detail, the system includes two main sub-modules: a panel detector and a part detector. The panel detector takes the input image, detects its keypoints and computes their SIFT descriptors. Then FLANN and RANSAC are used to match the content of the input image with the templates representing the panels. Templates consist in images where the panel is open and clearly visible; a polygonal approximation of the boundary of each panel have been manually outlined: only the keypoints detected inside the boundary dilated by 20% are used for the detection. Template images, boundary and

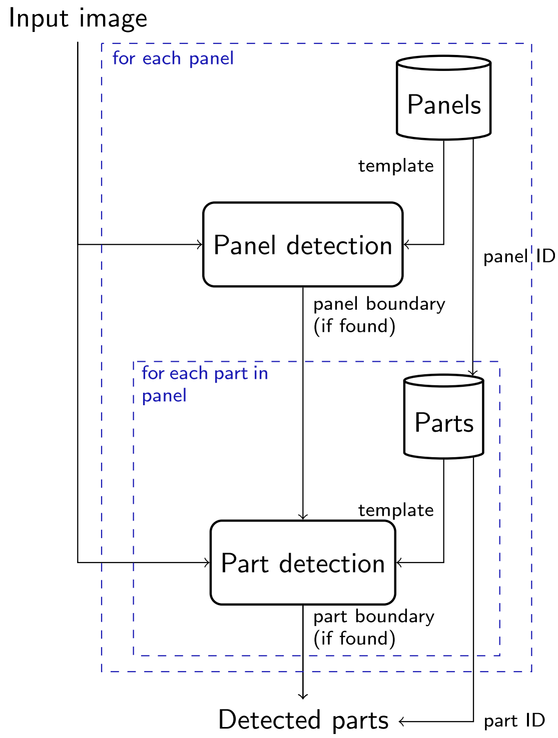


**Fig. 7.** Scheme of the system for the automatic detection of panels and mechanical parts.

keypoints are precomputed and stored in a database. Beside the criteria used by the RANSAC method, to reduce the amount of false positives additional heuristics have been introduced. These heuristics are based on the reprojection of the polygonal contour from the template image to the input image. Such a reprojected contour is obtained by applying Eq. (3) to the vertices of the contour on the template image. Candidate detections are discarded if one of the following conditions does not hold true:

- the shape of the reprojected contour must be convex;
- its baricenter must fall within the boundaries of the input image;
- the transformation must not imply an inversion of the orientation of the shape (i.e. if the vertices of the original contour have been given in a counterclockwise order, then they must still be in a counterclockwise order after the reprojection).

When a panel is found, the boundary of the corresponding template is backprojected on the input image. The keypoints inside the backprojected boundary are passed to the part detection module. Then, the templates representing the mechanical parts that are known to be located in the panel are matched against the input image with the same approach used to detect the panels. A graphical overview of the system is depicted in Fig. 7.

## 4. Experimental results

To assess the performance of the proposed system we conducted an experimentation on a set of images portraying a selection of panels and mechanical parts. More in detail we considered 248 images portraying 20 different mechanical parts located in eight different panels. The parts have been selected by experts from Alenia-Aermacchi to be representative of the elements involved in various kind of maintenance operations.

We used some additional images to provide the panels and parts ground-truth. Each panel and part has been manually annotated by drawing a polygon around the region of interest. The final ground-truth includes a very low number of examples of both panels and parts (from 1 to 5). The system has been developed in C/C++ and it has been embedded in the smart maintenance system as web service.

The computational cost of the visual recognition part is low. The time required to recognize a single mechanical part is about 500 ms on a Linux Ubuntu machine equipped with an Intel Core i7-4790 (3.60 GHz $\times$ 8 CPUs) and a 16 GB RAM. This time increases as the number of mechanical part samples used as ground truth increases. In this case the number of possible comparisons increases and this influences the computational time. However, we also noticed that the use of a high number mechanical part samples as ground truth does not influence too much the accuracy of the visual recognition module and it is relevant only for those parts that can be captured under large variations of the view point. For this reason we decided to keep this number very low.

As performance measures for the evaluation of the mechanical parts recognition, we considered the detection rate (i.e. the fraction of instances of the part that have been correctly localized by the system) and the number of false positives (i.e. the number of wrong

**Table 1**
Performance of the proposed system on 248 images depicting eight panels and 20 different mechanical parts.

| # | Part name | Panel | Detections | No. of instances | Detection rate (%) | False positives |
|---|---|---|---|---|---|---|
| 1 | Get home display | A | 41 | 62 | 66.1 | |
| 2 | Oxygen/air refueling control panel | A | 58 | 62 | 93.5 | |
| 3 | Engine/fuel display panel | A | 22 | 37 | 59.5 | |
| 4 | Flight control computer (A &D) | B | 19 | 21 | 90.5 | |
| 5 | Store management Computer | B | 15 | 21 | 71.4 | |
| 6 | Lefas bevel gearbox (left side) | B | 13 | 16 | 81.2 | |
| 7 | Flight control computer (B &C) | C | 30 | 31 | 96.8 | |
| 8 | Miscellanea computer assay | C | 31 | 31 | 100.0 | |
| 9 | Lefas bevel gearbox (right side) | C | 11 | 14 | 78.6 | |
| 10 | Rudder control module | D | 24 | 25 | 96.0 | 1 |
| 11 | Servo actuator horizontal tail | D | 19 | 20 | 95.0 | 2 |
| 12 | Valve fuel shut off DC motor operated | E | 39 | 43 | 90.7 | |
| 13 | Crash survival memory unit | E | 32 | 41 | 78.0 | |
| 14 | Formation light power supply | E | 13 | 28 | 46.4 | |
| 15 | Hydraulic filter package (left side) | F | 18 | 19 | 94.7 | |
| 16 | Hydraulic filter package (right side) | F | 19 | 21 | 90.5 | 2 |
| 17 | Navigation light power supply | G | 22 | 22 | 100.0 | |
| 18 | VOR ILS receiver | G | 21 | 22 | 95.5 | 1 |
| 19 | Valve fuel shut off DC motor operated | G | 19 | 22 | 86.4 | |
| 20 | Container assy fire extinguisher | H | 8 | 8 | 100.0 | |
| | Total | | 474 | 566 | 83.7 | 6 |


**Fig. 8.** One example of correct detections for each of the eight panels (A–H from top to bottom). The identifier and the boundary of the detected parts are displayed on top of the input images.

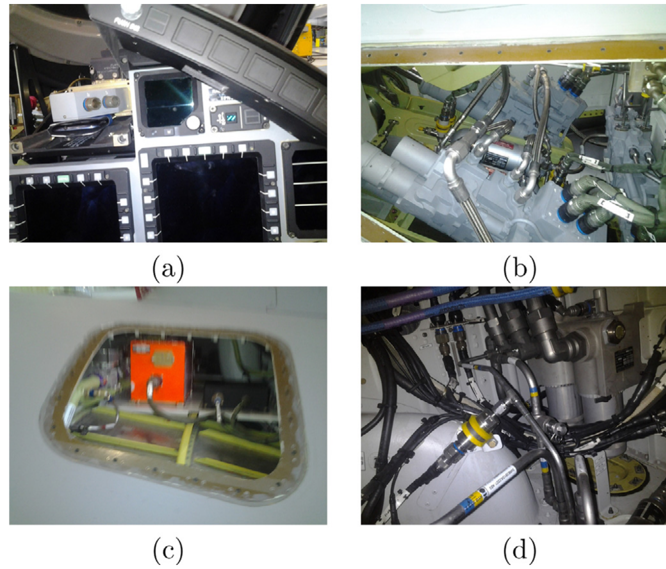


**Fig. 9.** Examples of factors causing missed detections: (a) reflections on a display; (b, d) occlusion of the part of interest; (c) motion blur. See Fig. 8 for an indication of the parts that the system was supposed to find.

detections found by the system, including the cases where the wrong ID has been assigned to a part, those where the correct part was detected but at a wrong location, and those where non-existing parts where found). In agreement with the requirements from Alenia-Aermacchi, and taking into account the nature of the use case described in Section 1, the parameters of RANSAC where tuned to produce as few false positives as possible, even at the cost of a decrease in the detection rate.

The results obtained are summarized in Table 1. For most parts the detection rate is quite high, and in three different cases (parts #8, #17 and #20) 100% of the instances have been correctly detected. On average the detection rate was of about 83.7%, Fig. 8 shows on example of correct detections for each panel. Most of the errors were missed detections of displays (#1 and #3), where highlights and reflections can disrupt the local descriptors. Another part with a low detection rate is the 'Formation Light Power Supply' (part #14), which has a very simple design: due to the low amount of details, very few keypoints are extracted for a possible matching; therefore, even a small amount of mismatches result in a missed detection. Other missed detections often depend on low-quality images (underexposed, with motion blur etc.). Fig. 9 shows a selection of missed detections.

We obtained a very small number of false positives (only six). All of them can be considered as partial detections, where the correct part was correctly identified, but badly localized. This happens when all the keypoints detected are clustered in a small region of the input image. In these conditions the estimation of the homography cannot be accurate enough, with the result that only part of the boundary is correctly located in the input image. Fig. 10 show the six false positives produced by the system.

The performance observed in the experimentation confirms that the proposed recognition module is suitable for the use case addressed in this work. In fact, the very small number of false positives (only six on a test set containing 566 detectable parts) ensures a very low chance of providing the wrong documentation to the operator that is using the application. More frequent are the cases in which the system fails to recognize the parts of interest (it

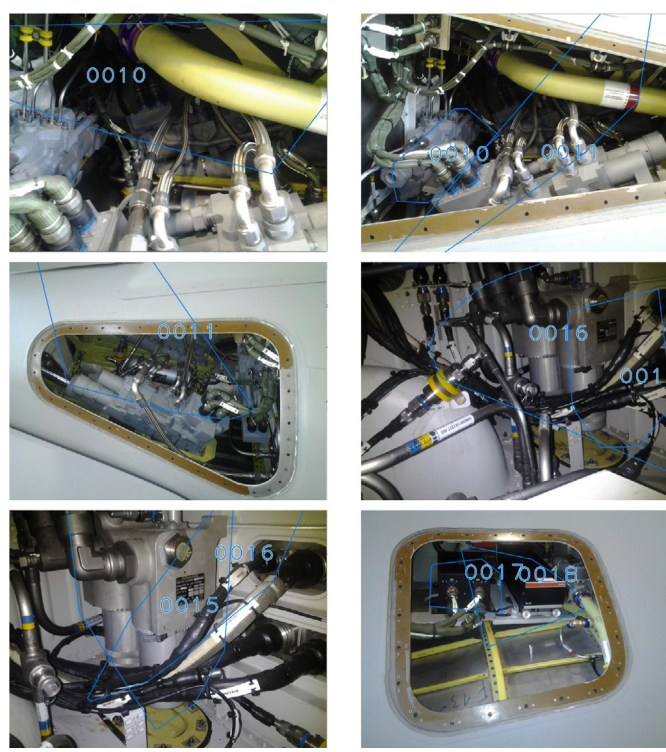


**Fig. 10.** The six false positives found by the system on the 248 test images.

happens in 16.3% of the cases), but this kind of failures has simply the effect of encouraging the operator to take better pictures. In fact, most of the missed detections are caused by an insufficient quality of the input image and we expect that the frequency of these cases will naturally drop as the operator learns to use the tool.

## 5. Conclusion

Maintenance and inspection operations represents a large portion of the costs related to an aircraft. Without compromising their reliability, they can be made more cost-effective by exploiting advanced information and communication technologies. Computer-vision techniques can be used to recognize and track a given mechanical part, making it possible to show related information to a technician on a suitable display. Previous works investigated the feasibility of computer-vision based techniques on a single maintenance operation, and without providing objective recognition performance.

In this paper we proposed a visual module that is able to recognize and to localize various mechanical parts of the aircraft. The module has been included in a prototype system designed for the smart maintenance of the Alenia-Aermacchi M346. The evaluation has been carried out on real aircrafts and addressed different kind of maintenance operations. We considered 248 images portraying eight different panels and 20 different mechanical parts taken under different imaging conditions: orientation, scale and illumination. The experimental results allowed us to verify the feasibility of our recognition module in a such a very challenging scenario.

## Acknowledgment

The authors would like to acknowledge the partners of the Smart Maintenance Project for the supporting of this research.

## References

[1] R. Saltoglu, N. Humaira, G. Inalhan, Scheduled maintenance and downtime cost in aircraft maintenance management, Int. J. Mech. Aerosp. Ind. Mechatron. Manuf. Eng. 10 (3) (2016) 580–585.
[2] X. Zhao, W.J. Verhagen, R. Curran, Aircraft Bi-Level Life Cycle Cost Estimation, IOS Press, 2015.
[3] Y. Asiedu, P. Gu, Product life cycle cost analysis: state of the art review, Int. J. Prod. Res. 36 (4) (1998) 883–908.
[4] M. Bazargan, Airline Operations and Scheduling, Routledge, 2016.
[5] D. Trentesaux, T. Borangiu, A. Thomas, Emerging ICT concepts for smart, safe and sustainable industrial systems, Comput. Ind. 81 (2016) 1–10.
[6] G. Dini, M. Dalle Mura, Application of augmented reality techniques in through-life engineering services, Proc. CIRP 38 (2015) 14–23.
[7] N.I. Adhani, R.D.R. Awang, A survey of mobile augmented reality applications, in: Proc. of the International Conference on Future Trends in Computing and Communication Technologies, 2012, 89–96.
[8] J. Campos, Development in the application of ICT in condition monitoring and maintenance, Comput. Ind. 60 (1) (2009) 1–20.
[9] G. Jo, K.-S. Lee, D. Chandra, C.-H. Jang, M.-H. Ga, RANSAC versus CS-RANSAC, in: Proc. of the Annual Conference on Innovative Applications of Artificial Intelligence, 2015, 1350–1356.
[10] M. Di Donato, M. Fiorentino, A.E. Uva, M. Gattullo, G. Monno, Text legibility for projected augmented reality on industrial workbenches, Comput. Ind. 70 (2015) 70–78.
[11] F. Archetti, G. Arosio, A. Candelieri, I. Giordani, R. Sormani, Smart data driven maintenance: Improving damage detection and assessment on aerospace structures, in: Proc. of IEEE Metrology for Aerospace (MetroAeroSpace), 2014, 101–106.
[12] M. Fiorentino, A.E. Uva, M. Gattullo, S. Debernardis, G. Monno, Augmented reality on large screen for interactive maintenance instructions, Comput. Ind. 65 (2) (2014) 270–278.
[13] F. De Crescenzio, M. Fantini, F. Persiani, L. Di Stefano, P. Azzari, S. Salti, Augmented reality for aircraft maintenance training and operations support, IEEE Comput. Graph. Appl. 31 (1) (2011) 96–101.
[14] G. Jo, K.-J. Oh, I. Ha, K.-S. Lee, M.-D. Hong, U. Neumann, S. You, A unified framework for augmented reality and knowledge-based systems in maintaining aircraft, in: Proc. of the Annual Conference on Innovative Applications of Artificial Intelligence, 2014, 2990–2997.
[15] G. Arosio, I. Giordani, L. Arieni, F. Archetti, Visual support and interaction for error prevention in aircraft maintenance, in: Metrology for Aerospace (MetroAeroSpace), 2014 IEEE, IEEE, 2014, May, pp. 372–376.
[16] R. Szeliski, Computer Vision: Algorithms and Applications, Springer Science & Business Media, 2010.
[17] A. Guiducci, Corner characterization by differential geometry techniques, Pattern Recognit. Lett. 8 (5) (1988) 311–318.
[18] K. Mikolajczyk, C. Schmid, Scale & affine invariant interest point detectors, Int. J. Comput. Vis. 60 (1) (2004) 63–86.
[19] D.G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2) (2004) 91–110.
[20] H. Bay, T. Tuytelaars, L. Van Gool, SURF: speeded up robust features, in: Proc. of IEEE European Conference on Computer Vision, 2006, 404–417.
[21] E. Rosten, T. Drummond, Machine learning for high-speed corner detection, in: Proc. of IEEE European Conference on Computer Vision, 2006, 430–443.
[22] T. Tuytelaars, K. Mikolajczyk, Local invariant feature detectors: a survey, in: Foundations and Trends® in Computer Graphics and Vision, vol. 3 (3), 2008, 177–280.
[23] K. Mikolajczyk, C. Schmid, A performance evaluation of local descriptors, IEEE Trans. Pattern Anal. Mach. Intell. 27 (10) (2005) 1615–1630.
[24] D.G. Lowe, Object recognition from local scale-invariant features, in: Proc. of the Seventh IEEE International Conference on Computer Vision, vol. 2, 1999, 1150–1157.
[25] M. Calonder, V. Lepetit, C. Strecha, P. Fua, BRIEF: binary robust independent elementary features, in: Proc. of IEEE European Conference on Computer Vision, 2010, 778–792.
[26] S. Leutenegger, M. Chli, R.Y. Siegwart, BRISK: binary robust invariant scalable keypoints, in: Proc. of IEEE International Conference on Computer Vision, 2011, 2548–2555.
[27] E. Rublee, V. Rabaud, K. Konolige, G. Bradski, ORB: an efficient alternative to SIFT or SURF, in: Proc. of IEEE International Conference on Computer Vision, 2011, 2564–2571.
[28] O. Miksik, K. Mikolajczyk, Evaluation of local detectors and descriptors for fast feature matching, in: Proc. of International Conference on Pattern Recognition (ICPR), 2012, 2681–2684.
[29] M. Muja, D. Lowe, Fast approximate nearest neighbors with automatic algorithm configuration, in: Proc. of the 4th International Conference on Computer Vision Theory and Applications, vol. 1, 2009, 331–340.
[30] C. Silpa-Anan, R. Hartley, Optimised KD-trees for fast image descriptor matching, in: Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, 2008, 1–8.
[31] B. Leibe, K. Mikolajczyk, B. Schiele, Efficient clustering and matching for object class recognition, in: Proc. of the British Machine Vision Conference, 2006, 789–798.
[32] R. Hartley, A. Zisserman, Multiple View Geometry in Computer Vision, Cambridge University Press, 2003.
[33] M.A. Fischler, R.C. Bolles, Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography, Commun. ACM 24 (6) (1981) 381–395.
[34] P.H. Torr, A. Zisserman, MLESAC: a new robust estimator with application to estimating image geometry, Comput. Vis. Image Understanding 8 (1) (2000) 138–156.
[35] O. Chum, J. Matas, S. Obdrzalek, Enhancing RANSAC by generalized model optimization, Proc. of the Asian Conference on Computer Vision, vol. 2 (2004) 812–817.
[36] O. Chum, J. Matas, Matching with PROSAC-progressive sample consensus, in: IEEE Conference on Computer Vision and Pattern Recognition, vol. 1, 2005, 220–226.
[37] S. Choi, T. Kim, W. Yu, Proc. of the British Machine Vision Conference, 2009, 81.1–81.12.