



## Automated exterior inspection of an aircraft with a pan-tilt-zoom camera mounted on a mobile robot

Igor Jovančević, Stanislas Larnier, Jean-José Orteu, Thierry Sentenac

### ► To cite this version:

Igor Jovančević, Stanislas Larnier, Jean-José Orteu, Thierry Sentenac. Automated exterior inspection of an aircraft with a pan-tilt-zoom camera mounted on a mobile robot. *Journal of Electronic Imaging*, 2015, 24 (6), pp.061110. 10.1117/1.JEI.24.6.061110 . hal-01351008

**HAL Id: hal-01351008**

**<https://hal.science/hal-01351008>**

Submitted on 2 Aug 2016

**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.

# Journal of Electronic Imaging

JElectronicImaging.org

## **Automated exterior inspection of an aircraft with a pan-tilt-zoom camera mounted on a mobile robot**

Igor Jovančević  
Stanislas Larnier  
Jean-José Orteu  
Thierry Sentenac

# Automated exterior inspection of an aircraft with a pan-tilt-zoom camera mounted on a mobile robot

Igor Jovančević,<sup>a,\*</sup> Stanislas Larnier,<sup>b,c</sup> Jean-José Orteu,<sup>a</sup> and Thierry Sentenac<sup>a,b,c</sup>

<sup>a</sup>Université de Toulouse, CNRS, INSA, UPS, Mines Albi, ISAE, Institut Clément Ader, Campus Jarlard, F-81013 Albi, France

<sup>b</sup>CNRS, LAAS, 7 Avenue du Colonel Roche, F-31400 Toulouse, France

<sup>c</sup>Université de Toulouse, UPS, LAAS, F-31400 Toulouse, France

**Abstract.** This paper deals with an automated preflight aircraft inspection using a pan-tilt-zoom camera mounted on a mobile robot moving autonomously around the aircraft. The general topic is image processing framework for detection and exterior inspection of different types of items, such as closed or unlatched door, mechanical defect on the engine, the integrity of the empennage, or damage caused by impacts or cracks. The detection step allows to focus on the regions of interest and point the camera toward the item to be checked. It is based on the detection of regular shapes, such as rounded corner rectangles, circles, and ellipses. The inspection task relies on clues, such as uniformity of isolated image regions, convexity of segmented shapes, and periodicity of the image intensity signal. The approach is applied to the inspection of four items of Airbus A320: oxygen bay handle, air-inlet vent, static ports, and fan blades. The results are promising and demonstrate the feasibility of an automated exterior inspection. © 2015 SPIE and IS&T [DOI: [10.1117/1.JEI.24.6.061110](https://doi.org/10.1117/1.JEI.24.6.061110)]

Keywords: inspection; airplane exterior; image processing; pan-tilt-zoom camera.

Paper 15507SSP received Jun. 26, 2015; accepted for publication Oct. 20, 2015; published online Nov. 30, 2015.

## 1 Introduction

Airplanes are periodically inspected, either on the airport tarmac between flights or in a hangar during maintenance operations. Today, these inspections are done by human operators, mainly visually and sometimes with the help of some inspection tools, for instance, to evaluate the criticality of a crack detected on the airplane fuselage. In order to make the inspection quicker, more exhaustive, and more accurate, as well as for traceability reasons, a multipartner's research project is being carried on in order to develop a mobile robot equipped with several optical sensors [cameras and three-dimensional (3-D) scanners] to perform an automatic inspection of an airplane.

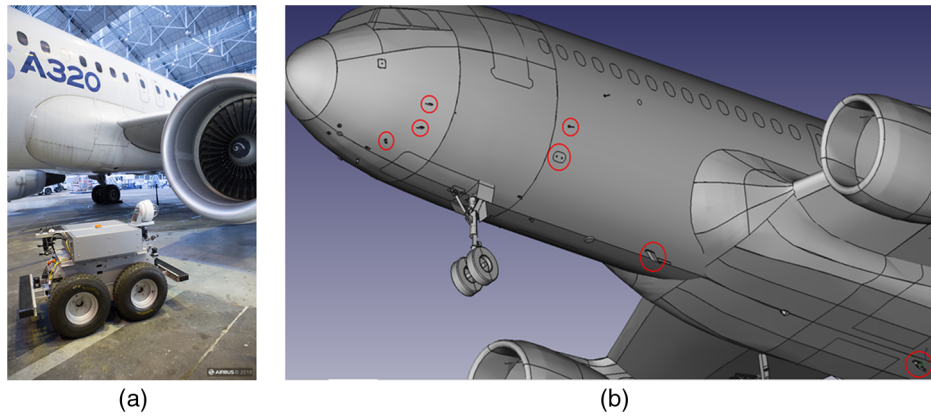
The approach chosen in the project is based on an autonomous mobile robot on the ground, capable of communicating with human operators and infrastructures around the aircraft. The collaborative robot (cobot) has to move autonomously between different checkpoints while interacting with pilots, workers, vehicles, and airport information systems. In addition to the long-range navigation task, at each checkpoint (around 20 for a preflight inspection), the cobot has to perform an inspection task of several items (probes, doors, etc.). Around 130 items to be inspected are identified for the whole aircraft. Automated image acquisition is provided by controlling sensors and pointing them toward the item to be inspected. There are two advantages in the present context. First, the navigation algorithms lead the robot to a checking position, which is localized in the aircraft model. Second, the 3-D model of the airplane with the items to be detected is known (Fig. 1). Therefore, it is possible, for vision inspection, to project the 3-D model onto the image plane to obtain the expected shape and position of the desired item.

Inspection is then performed by processing the pan-tilt-zoom (PTZ) camera images and 3-D point clouds provided by a 3-D scanner. For each item, integrity or right configuration should be verified. Accordingly, one of three general strategies is then chosen and applied: (A) single-view, two-dimensional (2-D) image processing; (B) model-based image processing (single-view image processing by using 3-D computer-aided design model of the element); and (C) processing of 3-D point clouds. The first strategy is preferred whenever possible, for its shorter time requirement. The second strategy is used when image processing is not sufficient, e.g., for inspecting probes. The third strategy is employed for detecting damages on the airplane skin (cracks, dents, and bumps).

Robotic airplane inspection from the ground, based on computer vision, is a specific application not studied enough. Previous efforts<sup>1-5</sup> were mainly focused on detailed airplane skin inspection with robots crawling on the airplane surface. In most cases, industrial automation systems<sup>6</sup> are used to inspect known objects at fixed positions with appropriate illumination necessary to extract meaningful features. In our application, however, sensors are mounted on the moving robot, and no additional illumination is used. Moreover, our intention is to enable the robot to cover as many items as possible from one position, i.e., to move as little as possible. Since 2-D image processing approaches are preferred, initial strategy is to assess performances of those approaches, and this paper is focused on presenting these results.

This paper is an extension of our work reported in Ref. 7. Compared to our previous work, we introduce here some improvements. The testing dataset is increased by varying brightness and contrast. Several texture-detector approaches are compared in order to achieve the best result in the background elimination step. We also suggest and compare

\*Address all correspondence to: Igor Jovančević, E-mail: [igor.jovancevic@mines-albi.fr](mailto:igor.jovancevic@mines-albi.fr)



**Fig. 1** (a) AIR-COBOT and A320 in the hangar (courtesy of Airbus). (b) Simplified three-dimensional model of the airplane.

alternative approaches for detecting items: edge-drawing circles (EDCircles) and low-contrast detector. The low-contrast detector is proposed as a solution for the inspection step as well, e.g., for the static port, due to the difference in shape in covered and uncovered cases. The detection of the spiral in the fan blade region enables locating problematic blades. We also introduce alternative line segment detectors (LSDs) with the needed postprocessing step of filtering and merging.

This paper is organized as follows. The dataset, the context, and our inspection process are explained in Sec. 2. Section 3 presents two strategies for item detection. The first one focuses on the shape edges and the second one focuses on the normals to the shape boundary. The numerical results on four items (oxygen bay handle, air-inlet vent, static ports, and fan blades) are given in Sec. 4.

## 2 Dataset, Context, and Process

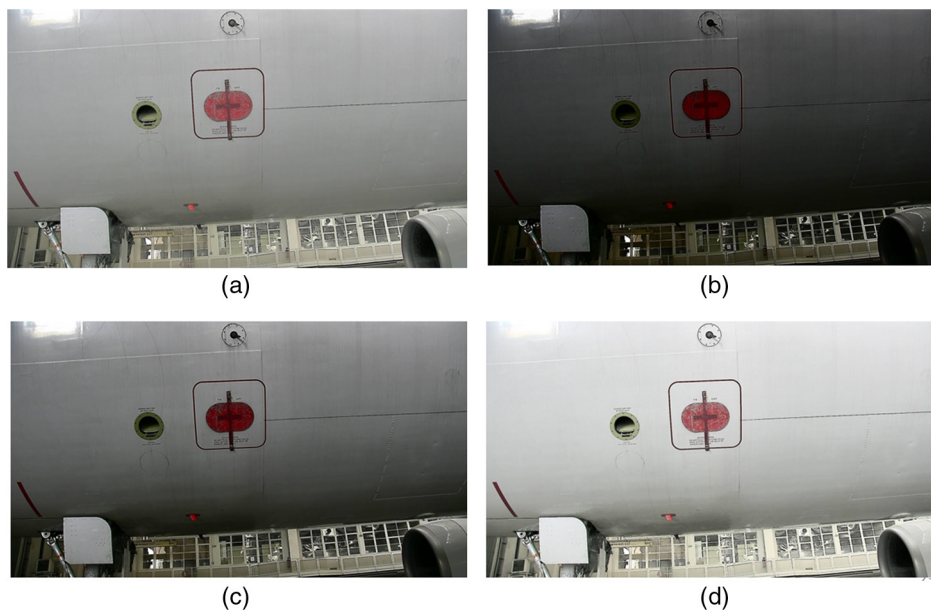
### 2.1 Dataset

The methods are tested on a dataset of real A320 airplane images acquired by PTZ camera in the Airbus hangar

(Fig. 1). The dataset is extended by programmatically increasing/decreasing brightness and/or contrast. For this purpose, the computationally efficient alternatives<sup>8</sup> of bias and gain functions proposed in Ref. 9 are computed. From each acquired image, we generate several new images with different levels of brightness and contrast (Fig. 2) in order to test the robustness of our approaches. We are simulating the conditions where the inspection will be done at the end of the day or in various weather conditions. The processed images have a resolution of  $1920 \times 1080$ .

### 2.2 Context and Process

For all the items, the approach comprises detection of region of interest (ROI) and the inspection, i.e., making decision. When possible, the presence of other detectable elements is exploited to narrow down the search area for an item to be inspected. Further, the expected position and size of the item in the image are also known. They are calculated by using a 3-D model of the airplane and the item within, as well as current position of the robot. Localization of the



**Fig. 2** (a–d) Image with different values of brightness and contrast.



robot is provided by simultaneous localization and mapping and visual servoing techniques. Due to the possible inclination of the robot, items can be rotated (Fig. 7). The PTZ camera allows us to correct the potential errors of robot poses at the checking points.

First, a blind pointing is made by knowing the airplane model and robot pose. Second, the item is detected in the low-zoom PTZ camera image (Fig. 3), and a visual servoing is performed to center the item. Lastly, the image is zoomed (Fig. 12) to obtain a better resolution, and the item state is verified by the inspection algorithm. Since the inspection is done only on high-zoom images, we can accept a certain level of imprecision in the region of interest (ROI) detection on low-zoom images.

The usual weakness of an inspection system is being specialized in inspecting one particular type of object. In that sense, each of the items on the airplane could be an isolated, specific, inspection problem. Our intention is to design more general strategies, applicable to a class of similar items by changing just the parameters and geometry. For instance, there are many rectangular items (doors) as well as circular shapes (vents, probe frames, etc.) on the airplane.

### 2.3 Background Removal

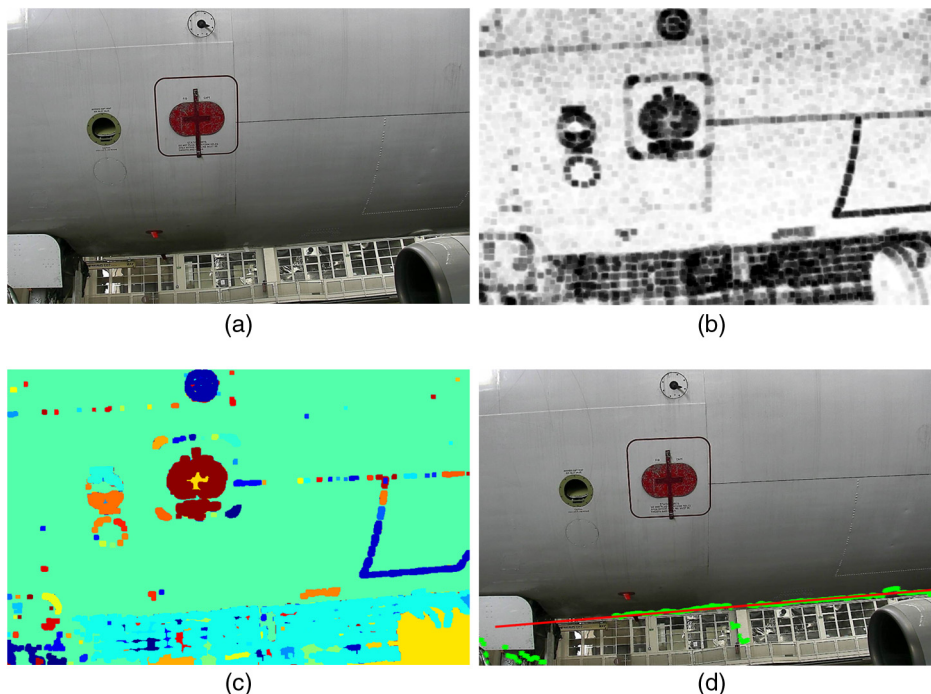
When the robot is placed on one of the sides of the airplane (Fig. 3), complex structures behind the aircraft, such as the hangar construction, are eliminated as highly textured areas close to the image borders. This way, only the upper, mainly uniform, part of the image, which corresponds to the airplane surface, is kept. This preprocessing step is performed by enhancing textured regions. The original, grayscale, smoothed image is convolved with  $5 \times 5$  Laws' kernel mask<sup>10</sup> obtained by multiplying two one-dimensional (1-D) vectors

$$\begin{bmatrix} 1 & -4 & 6 & -4 & 1 \end{bmatrix}^T \times \begin{bmatrix} -1 & 0 & 2 & 0 & -1 \end{bmatrix}.$$

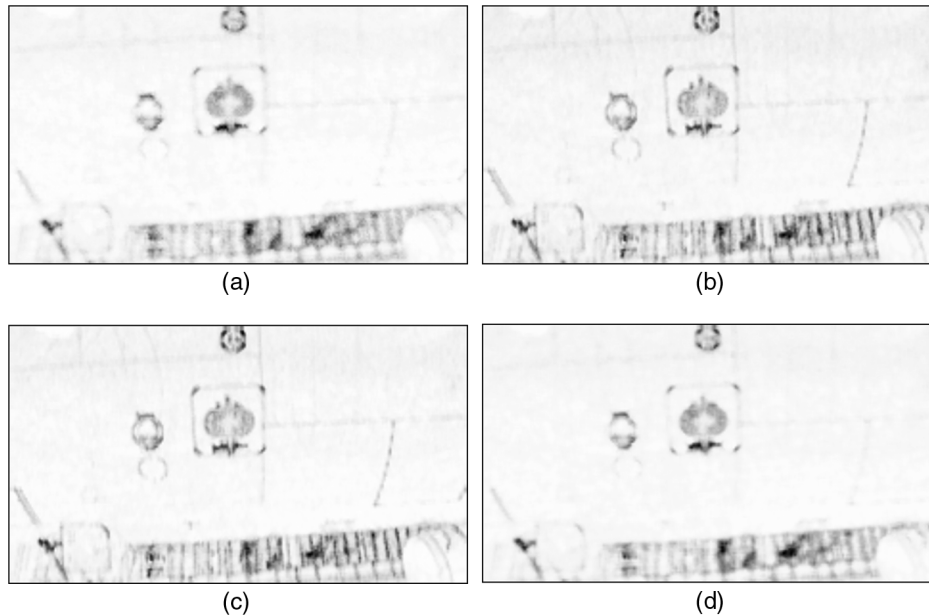
On the resulting image, a windowing operation of neighborhood averaging followed by morphological dilation and contrast enhancement is then applied [Fig. 3(b)]. Further, region-growing segmentation is performed on the texture image [Fig. 3(c)]. For locating the border between the airplane and the background, we rely on the fact that this border is almost a straight line and that the airplane is in the upper part of the image. By employing the random sample consensus technique<sup>11</sup> for removing outliers, we are fitting a straight line to the points obtained as peaks of the regions emerging from lower border of the image [Fig. 3(d)]. Finally, the image region below the fitted line is eliminated in further processing.

Many other texture approaches could be used.<sup>12,13</sup> Figure 4 presents four results with the texture-transform operator using a square window size of 32 pixels.<sup>14,15</sup> These results are based on the same method. Given an image, consider a  $w \times w$  square neighborhood of a pixel and let its gray values define the matrix  $W$ . Like any square matrix, it can be converted into a diagonal or triangular matrix, by pre- and postmultiplication of suitable matrices. The numbers  $\alpha = (\alpha_1, \dots, \alpha_w)$  in the diagonal of the matrix are numbered in decreasing order of magnitude. For example, these numbers can be computed with different decompositions of  $W$ : singular value decomposition, eigenvalues decomposition, decomposition into a product of an orthogonal and an upper triangular matrix (QR decomposition), or lower upper decomposition. A set of  $w$  numbers describes each pixel in the image. The texture transform is defined as

$$\phi(l, w) = \sum_{k=1}^w \|\alpha_k\|^z, \quad 1 \leq l \leq w.$$



**Fig. 3** Background removal. (a) Original image, (b) texture image after dilation and contrast enhancement (textured regions are darker), (c) region-growing segmentation on texture image, and (d) line fitted on the points that are on the top of the regions emerging from the low border of the image.



**Fig. 4** Examples of texture detection with the texture-transform operator. (a) Texture-transform result with eigenvalues, (b) texture-transform result with lower upper, (c) texture-transform result with QR, and (d) texture-transform result with singular value decomposition.

Each pixel is represented by a single number, which depends on three parameters. The number  $w$  obviously corresponds to the scale. The two second parameters  $l$  and  $\gamma$  are chosen empirically,  $l = \lfloor w/2 \rfloor$  and  $\gamma = 2$ , the same values as in Targhi's work.<sup>15</sup> Since texture is a local rather than a point-wise feature, it is not necessary to compute the descriptor at each pixel. A spacing parameter  $\delta$  set to 8 is used in the presented results. To avoid undesirable effects, the image patch is rotated by an angle  $\theta = 22.5, 45, 67.5$  and only the minimum response over orientation  $\theta$  is kept.<sup>15</sup>

The results are quite similar among these approaches. The dataset needs to be extended with other hangar and outside acquisitions to have a better comparison between them. So far, the best approach, in terms of computing time, is Laws' kernel mask.

### 3 Proposed Strategies for Item Detection

The two main strategies for airplane item detection are proposed. The items to be inspected are shown in Sec. 4. The first strategy is based on the shape edges. It is computationally more efficient and strongly relies on the geometric feature extraction techniques. Algorithms based on a Hough transform (HT) provide good results in high enough contrast conditions. Another approach, EDCircles, is also presented. This algorithm is efficient in low contrast as well.

The second strategy is based on the normals to the shape boundary. It could be seen as a template-matching-based approach, and it is robust to low-contrast conditions (Fig. 17), because it is using a normalized image gradient. However, it is less computationally efficient than the previous one.

#### 3.1 Detectors Based on Shape Edges

##### 3.1.1 Hough transform

Originally published in Ref. 16, the Hough transform (HT) with its numerous variants is traditionally used technique for extracting geometric features, particularly straight lines,

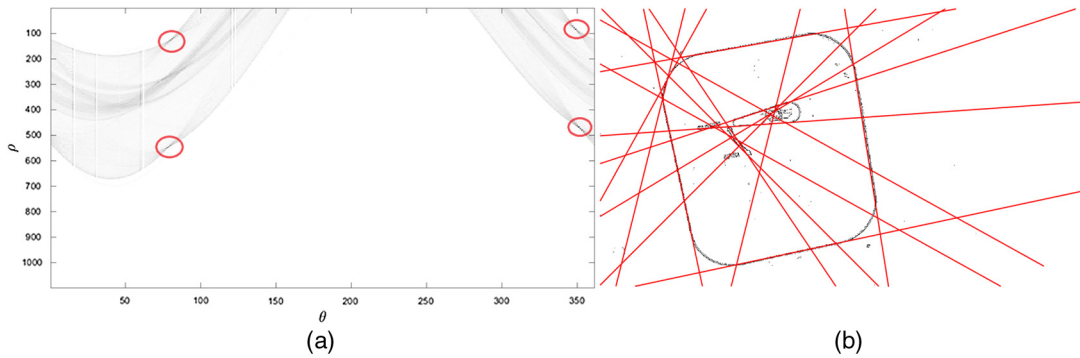
circles, and ellipses, from an edge image. In the case of straight lines, each possible line is represented by its distance from the origin  $\rho$  and the angle formed by the line normal to the positive part of the  $X$ -axis  $\theta$ .<sup>17</sup> Note that  $(\rho, \theta)$  are polar coordinates of the line point  $(x, y)$ , which is closest to the origin. The relation is given by a known Eq. (1):

$$\rho = y \sin(\theta) + x \cos(\theta). \quad (1)$$

The line parameter space (also known as the Hough space) is further defined as a matrix, whose rows and columns are denoted with  $\rho$  and  $\theta$  values, respectively. Therefore, each cell in this accumulator array determines one straight line. The size of the accumulator depends on the predefined quantization step for possible  $\rho$  and  $\theta$  values. Firstly, this matrix is initialized with zeros. Further, each edge pixel increases by one all the cells representing lines which pass through this pixel. This process is called "voting" and an example of the resulting accumulator array can be seen in Fig. 5(a). This simple case with no other objects in the scene clearly shows four peaks (four darkest pixels) in the array, which correspond to four lines of our rectangle. After all the edge pixels "voted," the parameter space is thresholded in order to detect the most prominent straight lines [Fig. 5(a)]. The detection result is sensitive to quantization step and the threshold applied on parameter space. In the case of a fine step or high threshold, there is a risk of missed detections while the coarse step and low threshold bring lot of false positives (FPs).<sup>18</sup> For these reasons, in our work, quantization step is empirically tuned according to our dataset, and the detection threshold is avoided by keeping a constant number of most prominent lines.

##### 3.1.2 Edge-drawing circles

Akinlar and Topal<sup>19</sup> have proposed a robust circle-detection method, named EDCircles, which works in a bottom-up



**Fig. 5** Extraction of straight lines with Hough transform (HT). (a) Hough space (accumulator array) and (b) lines obtained after thresholding Hough space.

manner. They start by extracting the set of connected edge segments by employing their edge segment detector.<sup>20</sup> Then, the line segments are fitted giving one or more line segments on each of the connected edge segments.<sup>21</sup> Further, circular arcs are searched as sequences of line segments, which form the angles in between certain thresholds. A circle is then fitted to each such an arc, and only arcs with a small least-square error are kept. Finally, the circular arcs with similar centers and radii are combined into circles.

### 3.2 Detector Based on the Normal to the Boundary

Since the aircraft inspection has to be performed indoors and outdoors, at any time of the day and with different weather conditions, the item-detection algorithms have to work even with low-contrast ROI. In difficult light conditions, the localization and orientation of the robot can be shifted a bit. However, it is possible to generate a set of potential shapes with different rotations and scales by projecting the 3-D model onto the image plane.

The low-contrast detector presented in this section is general and depends on the normal of the boundary shape. It has been used in many different contexts, such as in nuclei extraction in biology<sup>22</sup> or automatic visual grading of grain kernels.<sup>23,24</sup> It works with high and low contrast and returns a value in the range  $[-1, 1]$ . Let  $\Omega$  be the image domain,  $u$  be the image,  $\omega$  be the shape to test,  $\partial\omega$  be its boundary, and  $x \in \Omega$  a location. The function is defined as follows:

$$E(\omega) = \frac{1}{|\partial\omega|} \int_{\partial\omega} \left\langle \frac{\nabla u(x)}{\sqrt{|\nabla u(x)|^2 + \varepsilon^2}}, n(x) \right\rangle dx, \quad (2)$$

where  $\langle \cdot, \cdot \rangle$  denotes the Euclidean scalar product,  $n(x)$  denotes the outward normal to  $\omega$  at location  $x \in \partial\omega$ , and  $\varepsilon$  is a regularization parameter that discards faint transitions.

The computation of the energy map is quite simple and fast. First, a filter corresponding to the operation is built. Second, a simple convolution with the image is performed. The speed comes from the fact that the convolution is the pointwise product in the Fourier domain. The number of filters depends on the set of possible shapes. The tests are performed starting with the most-expected shape to the less-expected ones, and the algorithm stops if a good match is found.

## 4 Results on Four Types of Items

In this section, the detection and inspection approaches will be explained in more detail depending on the item to be inspected. Numerical results on inspecting four different types of items will be reported. The detection phase is evaluated by counting images on which the approach was able to isolate ROI successful enough that the zooming is possible. Inspection approaches are evaluated by calculating the false positive rate (FPR) and false negative rate (FNR). The input image in terms of the tested approach is classified as FP, in the case when the defect is not present and the approach reported a defect. It is classified as false negative (FN), in the case when there is a defect and the method reports no defect. FPR is the ratio between number of FP images and the total number of images with no defect. This is the measure of false alarms

$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}}.$$

FNR is the ratio between the number of FN images and the total number of images with defects. This is the measure of missed defects and is considered critical in our application

$$\text{FNR} = \frac{\text{FN}}{\text{FN} + \text{TP}},$$

where TN is the number of images with no defect on which the method reported no defect, and TP is the number of images with defect on which the method reported a defect.

FNR increases when some defects are missed. Our priority is to achieve a very low FNR; ideally it should be zero. FPR is a measure of false-alarm rate, and it increases when our system reports nonexistent defects. FPR is less critical than FNR, but it is also desired to have it low.

### 4.1 Oxygen Bay Handle

#### 4.1.1 Objective

The objective is to check if the handle is latched [Fig. 6(a)]. The problem of ROI detection is to a certain degree similar to the problem of edge-based door detection<sup>25–28</sup> with a monocular camera. Neither of the cited methods is fully applicable to our problem. Reference 25 relies on corners and our corners are rounded, while Refs. 26–28 are not invariant to the rotation of a door.



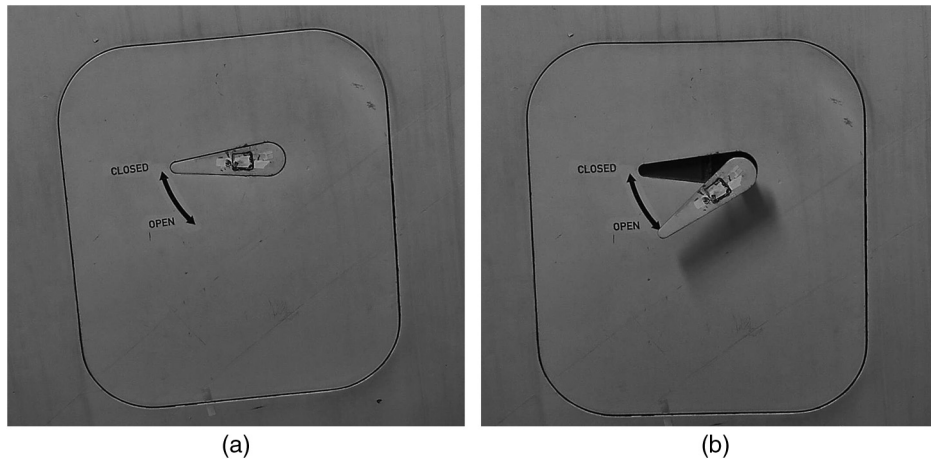


Fig. 6 Oxygen bay. (a) Latched and (b) unlatched.

#### 4.1.2 ROI detection

On the edge image, we apply HT<sup>17</sup> in order to detect the straight lines. HT suffers from computational complexity, but in our case, the edge set is not large. Since the scene can be complex, we allow detection of many lines in order not to miss some of the lines we are seeking. In order to filter the HT result, clusters of lines that are similar in terms of both  $\rho$  and  $\theta$  are identified, and only one line from each cluster is preserved. The resulting set of lines in the simple case is shown in Fig. 5(b).

Among many detected lines, the challenge is to pick four lines that bound the oxygen bay. First, many of the four lines' (two pairs) candidates are selected such that within each pair two lines are close to parallel, and between two pairs, lines are far from parallel (ideally perpendicular). We can safely assume that perspective distortion of the airplane surface is negligible, i.e., we can count to a certain degree on affine invariants such as parallelism of lines. The weakness of the HT is that it returns rays and not continuous segments, neglecting the gaps between the parts of a ray. Therefore, it can happen that we obtain a ray that is constituted by two or more distant-aligned segments [Fig. 7(b)]. To avoid this, once the four rays and their intersections are detected, segments that correspond to rectangle sides are checked for the "fill-ratio measure" similar to Ref. 25. The fill ratio of a segment is a measure of how much of the segment is supported by edge pixels. We are interested only in candidates whose all four segments are well supported by the edges [Fig. 7(a)], and we reject those whose segments have big gaps

[Fig. 7(b)]. When calculating the measure, we take into account only the middle part of the segment, having in mind that the corners are rounded; hence, the end parts of the segment are not expected to be supported by the edges.

After the fill-ratio criteria are verified, the knowledge about the real-world "aspect ratio" of the sides of the rectangle is used to find the candidate whose sides are closest to this ratio. Finally, the ROI is extracted as an area bordered by the four rays [Fig. 7(a)].

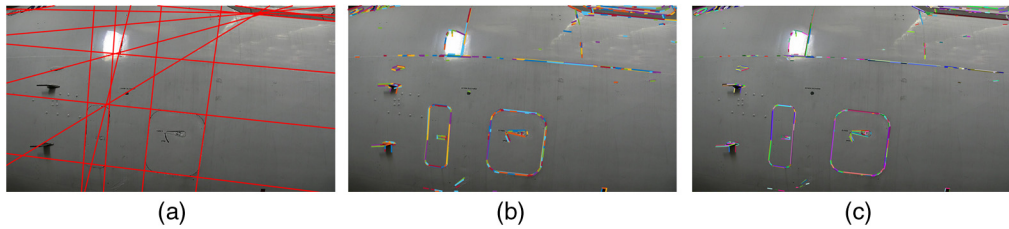
Another considered strategy is to start with one of the LSDs present in the literature, such as LSD<sup>29</sup> and edge-drawing lines (EDLines).<sup>21</sup> Outputs of these methods demand a sensitive postprocessing step of "filtering" due to the large set of detected segments (second and third image of Fig. 8). First, segments that are aligned (approximately belong to one ray) are identified, and the average ray is calculated. Then, this set of aligned segments is projected onto the average ray, and the projections are divided into clusters of overlapping (or nearby) segments. The segments of each such cluster are finally merged into one segment. This step is shown in Fig. 9. This approach is being considered mainly due to the FP control provided by proposed detectors and the fact they are parameterless, as opposed to traditional HT.

In our work, we have chosen to employ the well-known HT technique. The HT approach was sufficient in our application due to the fact that in our scenario, the position of the robot with respect to the airplane is expected to be nearly the same each time when inspecting one item. Therefore, the scene does not change significantly from case to case, and it



Fig. 7 Two candidates with their four segments (red). (a) Good candidate for being the oxygen bay—all four segments are well supported by the edges. (b) Bad candidate for being the oxygen bay—three out of four segments are poorly supported by the edges.





**Fig. 8** Comparison of line detectors. (a) HT result, (b) line segment detector (LSD) result, and (c) EDLines result.

was possible to tune the parameters. As it can be seen in Fig. 8(a), searched lines are clearly identified by HT, even though some FPs are present as well.

Like the background subtraction, other approaches could replace the one currently used for the line-features extraction. Further experiments with different aircraft and light conditions will be needed before determining the best one.

#### 4.1.3 Inspection

The smaller ROI (handle) is further isolated in the zoomed image as a convex hull of the set of edge points in the upper half of the big ROI [Figs. 10(a) and 10(c)]. This small ROI is binarized by using the well-known Otsu method<sup>30</sup> for adaptive thresholding. A clue used for making the decision (latched/unlatched) is the ratio between the area of the black region and the area of the whole smaller ROI. It is low in the latched case [Fig. 10(b)] and high in the unlatched case [Fig. 10(d)].

#### 4.1.4 Evaluation

The detection approach was tested on a dataset of 128 images of four different kinds of rectangular doors with different dimensions. It was successful in 96% of the cases. The inspection method was evaluated on a dataset of 122 images (86 negative and 36 positive cases) of four different kinds of

doors, each having a different position of the handle. It has shown accuracy with 0% FNR and 4.6% FPR.

### 4.2 Air-Inlet Vent

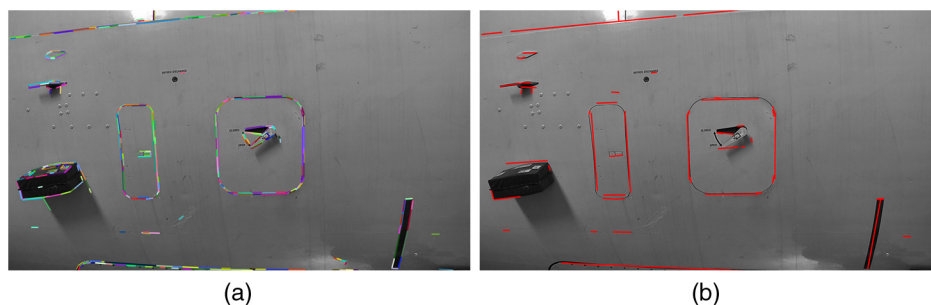
#### 4.2.1 Objective

The objective is to verify that the vent is “closed” [Fig. 12(a)]. A challenging case is when the inlet is “partially open” [Fig. 12(c)].

#### 4.2.2 ROI detection

ROI detection is a known image-processing problem of ellipse detection. The original HT-based method<sup>17</sup> suffers from inefficiency due to the five-dimensional (5-D) accumulator array (five parameters of an ellipse). Many studies have been conducted on improving the randomized HT approach for ellipse detection in terms of performance. In that sense, in Ref. 31, the authors apply HT on low-resolution images in an image pyramid data structure and then gradually increase resolution until the original resolution is reached.

The method proposed in Ref. 32 saves computational time by simplifying the problem to a 1-D parameter space instead of 5-D one used by the generalized HT. By employing this algorithm, we could detect the air-inlet vent with 95% accuracy on our 23-image dataset. Nevertheless, this result is



**Fig. 9** Postprocessing filtering step. (a) Output of LSD segment detector and (b) set of segments after filtering and merging. In this case, only nearly horizontal or vertical segments are kept after filtering and merging.



**Fig. 10** Oxygen bay—region of interest (ROI) binarization by Otsu method. (a) Latched case: ROI, (b) latched case: binarized ROI, (c) unlatched case: ROI, and (d) unlatched case: binarized ROI.

obtained after significant narrowing down the search area by detecting other items such as the neighboring static port (Sec. 4.3). Also, the size of the ellipse's major and minor axis had to be preset according to our dataset.

Therefore, we have chosen the recent EDCircles method proposed in Ref. 19, which has shown fewer false alarms [Fig. 11(c) and 11(d)] and less missed detections [Fig. 11(a) and 11(b)] while being faster as well. Another advantage of this approach is that it is parameter-free (as opposed to HT-based methods). It is capable to detect ideal circles and near-circular ellipses; however, it is not suitable for detecting strongly elliptical shapes. Since the position of the robot is such that the circles are not very distorted, we can accept this constraint.

Another strategy that is considered involves fitting the ellipses on the edge points. This approach contains two steps: preprocessing the edge image in order to link cut-edge sets and then running one of the ellipse-fitting methods<sup>33</sup> on each set of connected edge points. However, this approach has been shown to be sensitive to the edge-linking step. Consequentially, often the ellipse is not fitted precisely because it used only part of the edge set. It is clear that fitting-based methods rely strongly on the edge-detection method, which provides input to the fitting.

After all the ellipses in the image are detected [Fig. 11(b)], by knowing the placement of elliptical items in the scene, and having an idea about the size of the items in the image, we search for the correspondences between items expected to be in the scene and detected ellipses.

#### 4.2.3 Inspection

In order to perform the inspection, we estimate the binarized vent region similarly as we did with the handle in Sec. 4.1. However, since the vent can only be “partially open” [Fig. 12(c)], the binarization result of the whole ROI could be ambiguous as a clue. Thereby, for a more detailed

inspection, a small sliding window is moved along the border of the ROI (Fig. 12), and the uniformity of the sliding window is estimated by the binarization.

In the case of the open vent, there are two different regions within sliding window, while for the closed vent, the sliding window is uniform.

#### 4.2.4 Evaluation

The EDCircles algorithm was evaluated on a dataset of 23 low-zoom images (Fig. 2). It detected the vent in 100% of the cases, together with other present ellipses. This result did not change even on the same dataset with very low-brightness conditions [Fig. 11(c) and 11(d)].

Inspection was evaluated on 32 high-zoom (Fig. 12) images (13 negative and 19 positive cases) of two different kinds of vents. On this dataset, our sliding window method performed with both FNR and FPR equal to 0%. The same result was obtained on the same dataset in low-brightness conditions.

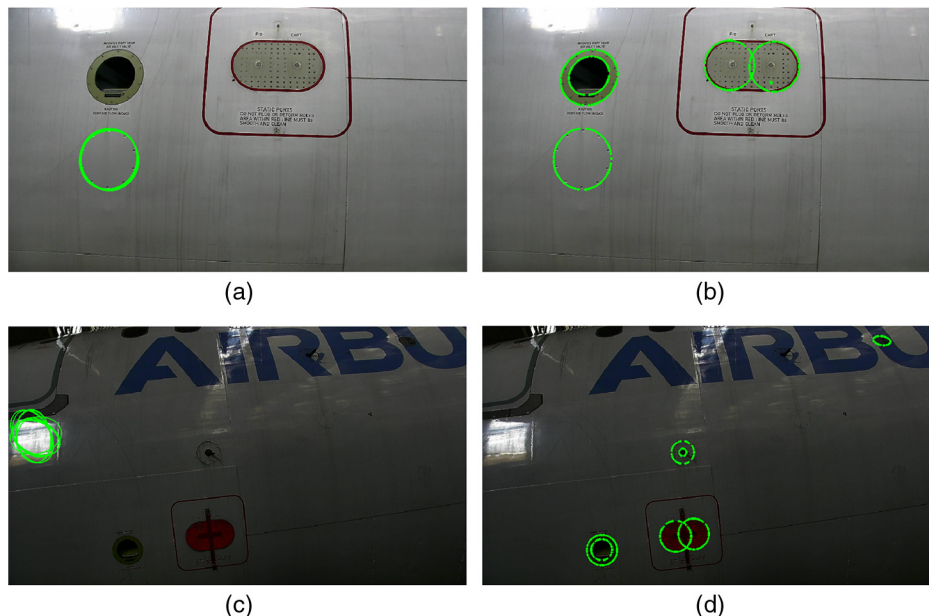
### 4.3 Static Ports

#### 4.3.1 Objective

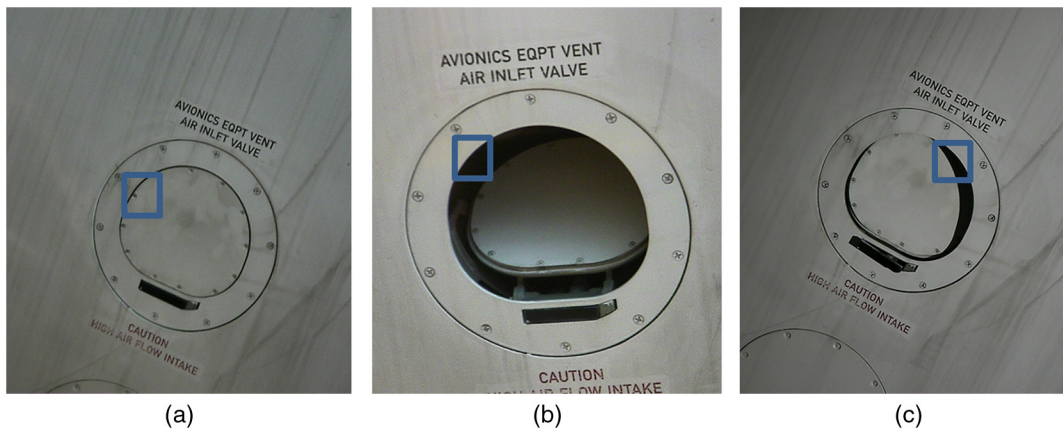
The aim is to verify that the protecting cover is removed from the static ports (Fig. 13).

#### 4.3.2 ROI detection

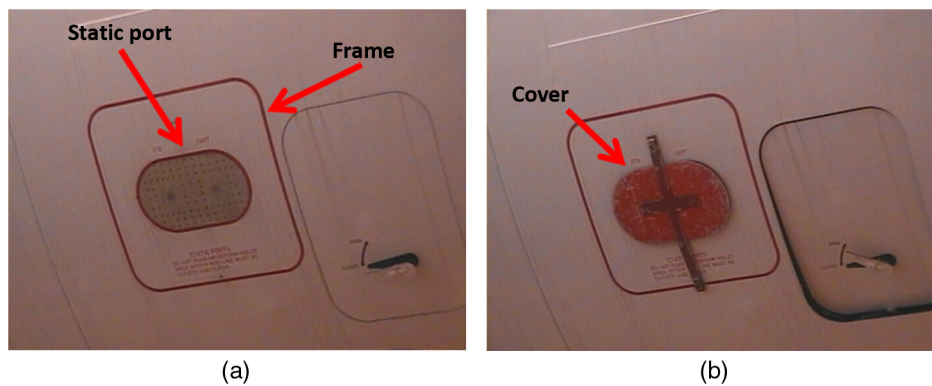
Since there are red regions on the image, the original RGB image is converted to a CIELAB color space. A\* channel of this color space is suitable for segmenting red regions. Further, from the input edge set, only edges corresponding to red regions are kept. The improvement can be seen in Fig. 14. On the reduced set of edges [Fig. 14(b)], the approach explained in Sec. 4.1 is employed to detect rectangular ROI.



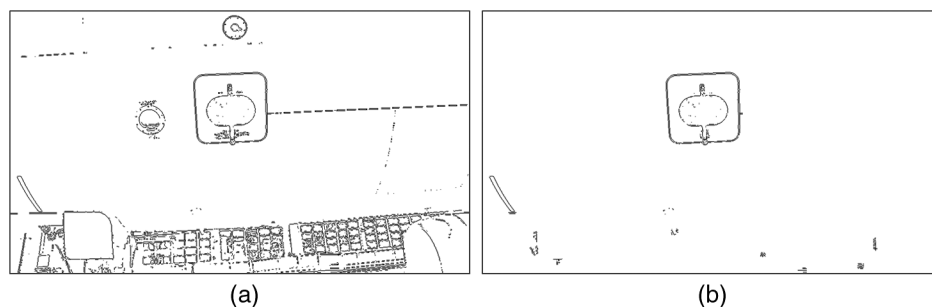
**Fig. 11** Comparison of the results one-dimensional (1-D) HT/EDCircles. (a) Standard conditions, 1-D HT result. (b) Standard conditions, EDCircles result. (c) Low-brightness conditions, 1-D HT result. (d) Low-brightness conditions, EDCircles result.



**Fig. 12** Air-inlet vent—sliding window inspection. (a) Closed vent (negative), (b) open vent (positive), and (c) partially open vent (positive).



**Fig. 13** First officer and captain static ports. Wrong situation is when protecting cover is still present. (a) Uncovered (negative) and (b) covered (positive).



**Fig. 14** Significant reduction of edge set by rejecting edges of nonred regions. (a) All the edges of the input image and (b) edges corresponding to red regions.

### 4.3.3 Inspection

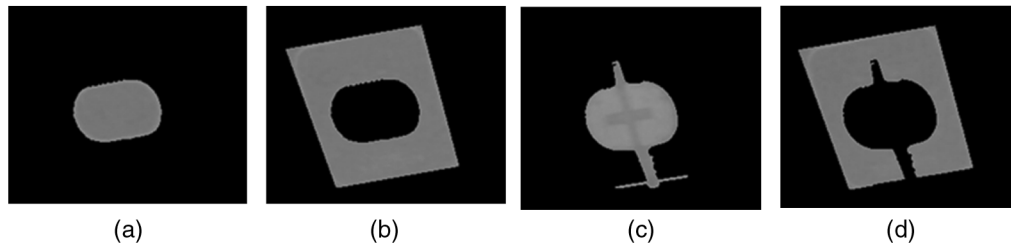
On the zoomed image, segmentation based on a region-growing technique is employed in order to segment the static port region, by setting the seed pixel in the mean position of the previously segmented rectangular ROI. A\* channel is used as a clue for making a decision as well. Namely, mean gray-level values of the regions shown in Figs. 15(a) and 15(b) are very close to each other (uncovered case); whereas in the case shown in Figs. 15(c) and 15(d) (covered case), they are different.

Another clue is convexity of the detected static port region, which is convex in the uncovered case [Fig. 15(a)] and concave in the covered case [Fig. 15(c)]. This convexity is estimated by the ratio of the area of the segmented region and the area of its convex hull.

### 4.3.4 Evaluation

The detection approach was tested on a dataset of 75 images of two different kinds of static ports, and it was successful in 97% of the cases. There was no occlusion of the red frame,





**Fig. 15** (a) Uncovered case: static ports region, (b) uncovered case: frame without static ports region, (c) covered case: static ports region, and (d) covered case: frame without static ports region.

but brightness and contrast are changed as explained in Sec. 2.1 (Fig. 2). As expected, failure occurred when the image is too dark because the red color information is lost in some parts of the frame.

The difficulty that occurred with darker images in the inspection step is that the region-growing process did not stop at the boundaries of the static port. This problem is solved by estimating the brightness of the image and tuning the stopping criteria for the region-growing segmentation (setting lower values in case of dark images). The inspection method was evaluated on a dataset of 96 images (37 negative and 59 positive cases) with two different cover shapes. Images are also changed in terms of contrast and brightness. The approach has shown accuracy with both FNR and FPR equal to 0%.

#### 4.3.5 Application of low-contrast detector

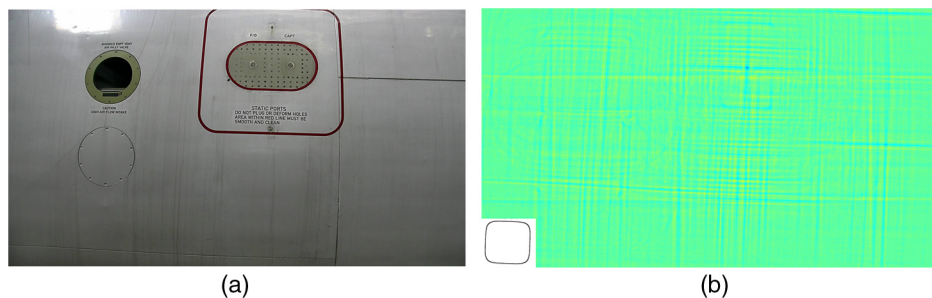
Since the line-based method (Sec. 4.1) relies on the edges and it assumes the item is fully visible, its weakness is identified in the case of occlusion (no edges) and in low-contrast conditions (weak edges). This could be caused, e.g., by an error in controlling the camera [Fig. 16(a)—occlusion] or by

specular reflection [Fig. 17(a)—low contrast]. In these cases, none of the four segments' candidates satisfied required conditions, so we employed a low-contrast detector, which is slower because it has to consider slight rotations and different scales of the element.

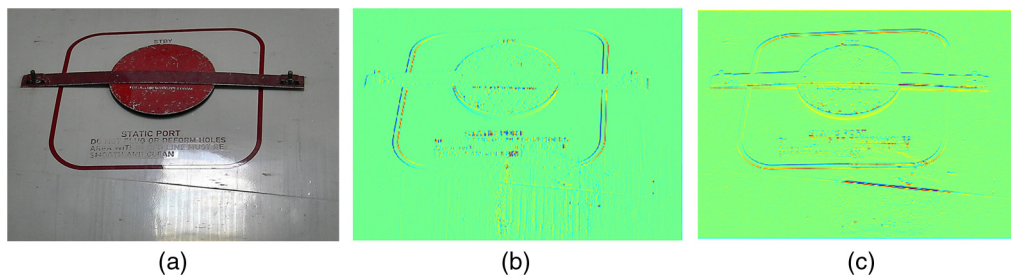
Figure 16 shows the interest of using the full shape of the target. The top line of the rounded square is out of the original image. The right image presents the associated energy map with the target shown in the bottom-left corner. The local minima of the energy map are in blue, and it is well centered on the static ports. The low-contrast detector is working with a partial target.

Figure 17 shows the interest of using low-contrast detector in the case of reflection on a portion of the static ports. The right corner of the rounded square is different from the other edges. In the red channel, it could be missed. One can note that the  $x$ - and  $y$ -derivatives of image gradient are still well defined even in the lighting zone (Fig. 17). It is why the low-contrast detector could be used when other detectors are unable to find a good candidate or when a too-bright zone is detected in the ROI where the item is supposed to be.

With the low-contrast detector, it is possible to define two filters to detect the state of the static ports. One adapted for



**Fig. 16** Example of camera-control problem on one of the static ports. (a) Original image and (b) energy map.



**Fig. 17** Example of light-condition problem on one of the static ports. (a) Original image, (b)  $x$ -derivative of image gradient, and (c)  $y$ -derivative of image gradient.



the covered case and one for the uncovered case. Figure 18 presents some detection results with a covered case in the first line and an uncovered case in the second line. In each energy map, the target shape is provided in the bottom-left corner: filled superellipse in the covered case and just superellipse in the uncovered case. The two filters allow identification of each of the two cases.

#### 4.4 Fan Blades

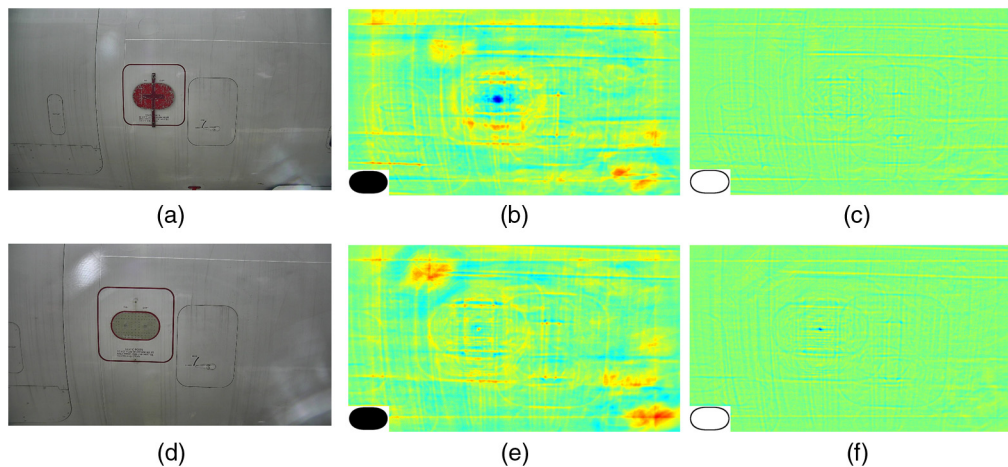
##### 4.4.1 Objective

In the blades' region, we should detect a large defect such as presence of visible object (Fig. 21) or big damage like a missing blade. The low-contrast detector is used to find the tip of the reactor. The detected transition is the one between

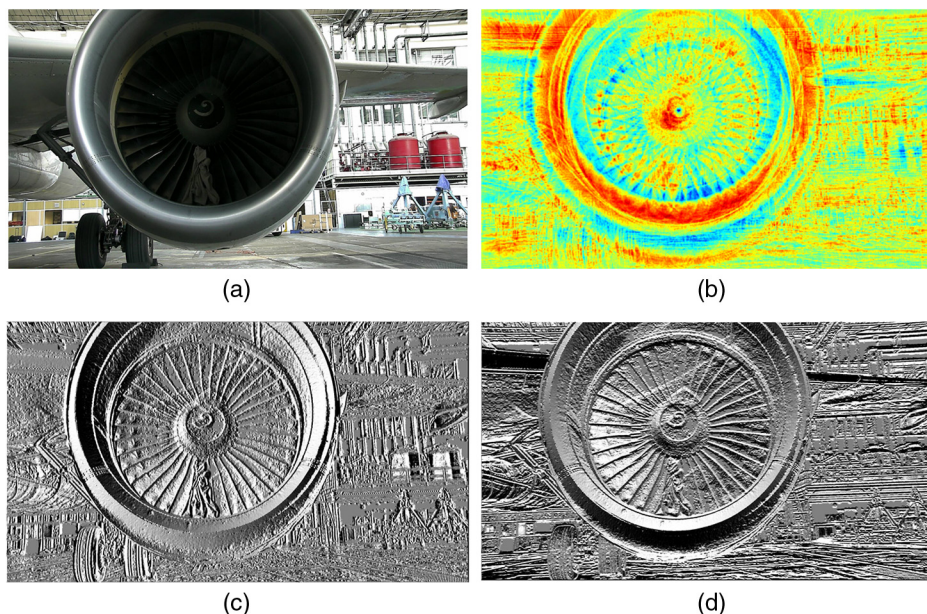
the black circular shape containing the spiral and the rest of the reactor. Figures 19(a) and 19(b) present the original image and the energy map, where the tip is well detected (dark blue region). Figures 19(c) and 19(d) provide the normalized image derivatives along  $x$  and  $y$ . After the tip is detected, the whole engine region is further calculated as a circle concentric to the detected tip.

##### 4.4.2 Inspection

Gray-scale intensity image is treated with histogram equalization technique for contrast enhancement, because blades are usually not well illuminated being placed deep inside the engine inlet. Further, we extract pixel values of many circles from the blades region (white in Fig. 22). Each circle is a 1-D signal that is extracted from the image and smoothed



**Fig. 18** Low-contrast detection algorithm applied to the static ports: (a–c) covered and (d–f) uncovered. (a–c) Covered static ports, energy map (covered test), and energy map (uncovered test). (d–f) Uncovered static ports, energy map (covered test), and energy map (uncovered test).



**Fig. 19** Detection of the tip of the reactor with the low-contrast detection algorithm. (a and b) Original image, energy map. (c and d)  $x$ -derivative of the normalized image gradient and  $y$ -derivative of the normalized image gradient.

with a Gaussian filter in order to eliminate small details. When there is no defect, a signal close to periodic is obtained [Fig. 20(b)]. However, in the case of defect, there is usually a breach of this periodicity [Fig. 21(b)].

By Fourier analysis, we compute the frequency of the highest magnitude component within the sliding window moved along the 1-D signal. In the favorable case, this frequency is constant as the sliding window is moved. In the case when a defect is present, frequency is changed when the sliding window is in the region of the defect. Nevertheless, some false alarms are present in the case of nonilluminated regions and in the regions where details behind the blades are visible [Fig. 22(b)]. To avoid these false alarms, only a defect that is present on many consecutive circles is reported as a foreign

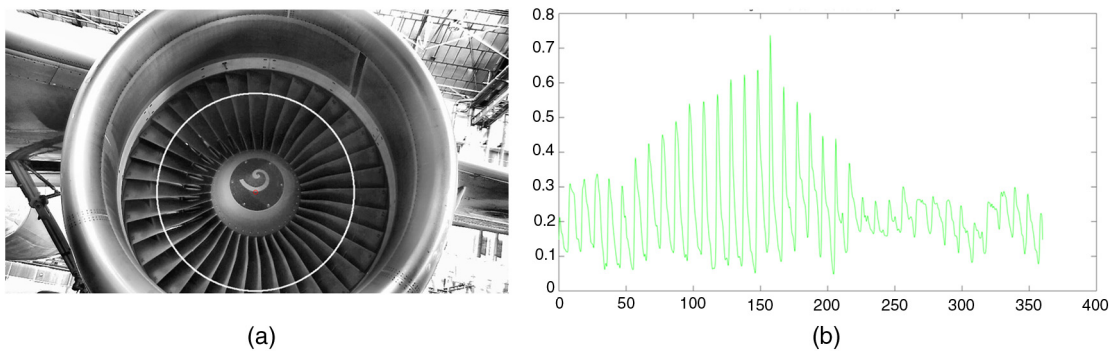
object or damage [Fig. 22(a)]. For this reason, at this time, the method is able to detect only large defects.

#### 4.4.3 Evaluation

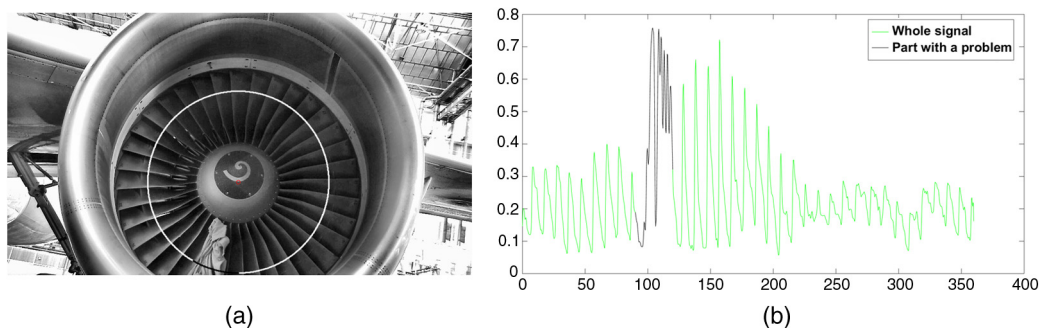
On a dataset of 49 images (24 negative and 25 positive cases), the inspection approach performed with 0% FNR and 16% FPR. It failed in the cases with blades being poorly illuminated and not zoomed enough.

#### 4.4.4 Spiral detection

When a defect is detected, it is useful to localize the problematic blades. The fan blades are numbered but they are not in fixed positions during every inspection session.



**Fig. 20** Example of 1-D signal read from the input image with no defect. (a) Extracted circle and (b) corresponding 1-D signal of pixel intensities on the circle.

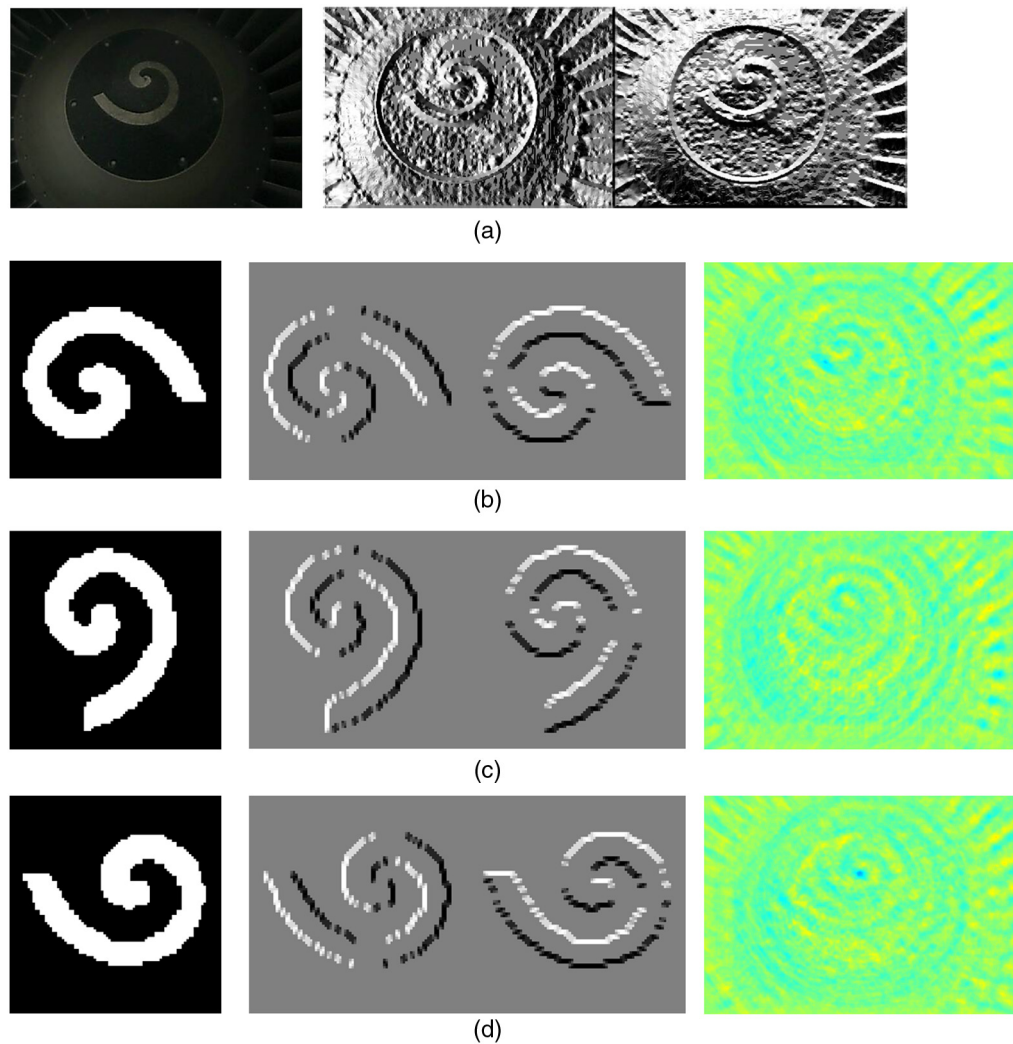


**Fig. 21** Example of 1-D signal read from the input image with defect. (a) Extracted circle. Black part is reported as defected. (b) Corresponding 1-D signal of pixel intensities on the circle with black part corresponding to the defect.



**Fig. 22** Fan blades—two cases to be distinguished and circular pixel sets which are analyzed. White: circles that are extracted from the image; black: parts where the approach reported a defect, i.e., breach of periodicity. (a) Defect—presence of an object and (b) no defect.





**Fig. 23** Low-contrast detection algorithm applied to the spiral at the tip of the aircraft engine. (a) Original image; x- and y-derivatives of the normalized image gradient. (b) Target 90; x- and y-normalized derivatives; energy map. (c) Target -90; x- and y-normalized derivatives; energy map. (d) Target 0; x- and y-normalized derivatives; energy map.

The white spiral at the tip helps to determine exactly the blades on which the defect is reported. If there is enough contrast, the spiral shape can be detected by a region growing or Otsu's binarization of the disk area. If these approaches do not work, there is probably not enough contrast and the spiral is detected by the low-

contrast detector. One can note from Fig. 23(a) that the spiral is well defined in the normalized image gradient. Further, several possible orientations of the spiral are tested and then the best match is kept. Figure 23 presents the spiral energy maps with a low-contrast detector and three orientations.

**Table 1** Evaluation results.

Item	Oxygen bay	Air-inlet vent	Static port	Fan blades
Detection dataset	128	23	75	—
Detection accuracy	96%	100%	97%	—
Inspection dataset	122 (86 negative 36 positive)	32 (13 negative 19 positive)	96 (37 negative 59 positive)	49 (24 negative 25 positive)
Inspection false positive rate	4.6%	0%	0%	16%
Inspection false negative rate	0%	0%	0%	0%

## 5 Conclusion

Several image-processing approaches, each solving a specific airplane-inspection problem, were developed and evaluated on the real airplane exterior images. On our dataset, without occlusions or contrast problems, the detection approaches showed accuracy of at least 95%, while inspection approaches could reach 0% FNR with acceptable false-alarm rates. Evaluation results are summarized in Table 1. In general, the whole process depends on the detection phase, which has been shown as more challenging than the inspection task. Since detection is performed on small-zoom images, knowledge about the presence of more than one item is integrated, and sets of neighboring items are detected instead of detecting one particular item. Notably, our method for red region detection has been shown to be quite robust; therefore, it is often employed to facilitate the detection of other items. In this paper, we presented inspection approaches for four representative types of items. Nevertheless, other items marked as highly important by human inspectors have been treated as well.

The testing dataset is obtained in a hangar with constant illumination. Some artificial illumination is added to make the dataset more representative. Furthermore, contrast and brightness are programmatically changed. Some proposed approaches work in low contrast, but more data are necessary to estimate their practical usability. Next, acquisition on the tarmac with various conditions, possibly with other airplanes present in the scene, and with various weather conditions are planned in the next few months.

Our dataset was limited to one plane in a hangar, but other validations are planned to be done. So far, the proposed methods were adapted for each item based on heuristics. A comparison with more general algorithms based on binary classification will also be investigated once sufficient training data are obtained.

## Acknowledgments

This work was 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 partners of the AIR-COBOT project (AKKA TECHNOLOGIES, Airbus Group, ARMINES, 2MoRO Solutions, M3 SYSTEMS and STERELA) are also gratefully acknowledged for their support.

## References

1. M. Siegel, P. Gunatilake, and G. Podnar, "Robotic assistants for aircraft inspectors," *IEEE Instrum. Meas. Mag.* **1**(1), 16–30 (1998).
2. M. Siegel, W. Kaufman, and C. Alberts, "Mobile robots for difficult measurements in difficult environments: application to aging aircraft inspection," *Rob. Auton. Syst.* **11**(3–4), 187–194 (1993).
3. I. L. Davis and M. Siegel, "Automated nondestructive inspector of aging aircraft," *Proc. SPIE* **2101**, 190–201 (1993).
4. P. Gunatilake and M. Siegel, "Remote enhanced visual inspection of aircraft by a mobile robot," in *Proc. 1998 IEEE Int. Workshop on Emerging Technologies, Intelligent Measurement and Virtual Systems for Instrumentation and Measurement (ETIMVIS'98)*, pp. 49–58, IEEE, Saint Paul, Minnesota (1998).
5. R. Mumtaz et al., "Computer aided visual inspection of aircraft surfaces," *Int. J. Image Process.* **6**(1), 38–53 (2012).
6. E. N. Malamas et al., "A survey on industrial vision systems, applications and tools," *Image Vision Comput.* **21**(2), 171–188 (2003).
7. I. Jovančević et al., "Automated visual inspection of an airplane exterior," *Proc. SPIE* **9534**, 95340Y (2015).

8. C. Schlick, "Fast alternatives to Perlin's bias and gain functions," in *Graphics Gems IV*, pp. 401–403, Academic Press Professional, Inc., San Diego, California (1994).
9. K. Perlin and E. M. Hoffert, "Hypertexture," in *SIGGRAPH Computer Graphics*, Vol. 23, pp. 253–262 (1989).
10. K. Laws, "Textured image segmentation," Ph.D. Dissertation, University of Southern California (1980).
11. M. Fischler and R. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM* **24**, 381–395 (1981).
12. K. I. Laws, "Rapid texture identification," *Proc. SPIE* **0238**, 376–380 (1980).
13. A. Materka and M. Strzelecki, "Texture analysis methods—a review," Tech. Rep., EU COST B11 report, Institute of Electronics, Technical University of Lodz, Poland (1998).
14. A. Targhi et al., "The eigen-transform and applications," *Lect. Notes Comput. Sci.* **3851**, 70–79 (2006).
15. A. Targhi, "The texture-transform: an operator for texture detection and discrimination," PhD Thesis, KTH School of Computer Science and Communication, Stockholm, Sweden (2009).
16. P. Hough, "Method and means for recognizing complex patterns," U.S. Patent No. 3,069,654 (1962).
17. R. O. Duda and P. E. Hart, "Use of the Hough transformation to detect lines and curves in pictures," *Commun. ACM* **15**, 11–15 (1972).
18. V. Patrăucean, P. Gurdjos, and R. Grompone von Gioi, "A parameterless line segment and elliptical arc detector with enhanced ellipse fitting," in *Computer Vision ECCV 2012*, pp. 572–585, Springer, Berlin, Heidelberg (2012).
19. C. Akinlar and C. Topal, "EDCircles: a real-time circle detector with a false detection control," *Pattern Recognit.* **46**(3), 725–740 (2013).
20. C. Topal and C. Akinlar, "Edge drawing: a combined real-time edge and segment detector," *J. Visual Commun. Image Represent.* **23**(6), 862–872 (2012).
21. C. Akinlar and C. Topal, "EDLines: a real-time line segment detector with a false detection control," *Pattern Recognit. Lett.* **32**(13), 1633–1642 (2011).
22. E. Soubières, P. Weiss, and X. Descombes, "A 3D segmentation algorithm for ellipsoidal shapes. Application to nuclei extraction," in *ICPRAM—Int. Conf. on Pattern Recognition Applications and Methods*, pp. 97–105, Barcelona, Spain, (2013).
23. P. Duboscq et al., "Deterministic method for automatic visual grading of seed food products," in *Int. Conf. on Pattern Recognition Applications and Methods*, SciTePress (2015).
24. P. Duboscq et al., "Automated visual grading of grain kernels by machine vision," *Proc. SPIE* **9534**, 95340H (2015).
25. Y. Tian, X. Yang, and A. Arditi, "Computer vision-based door detection for accessibility of unfamiliar environments to blind persons," in *Proc. of the 12th Int. Conf. on Computers Helping People with Special Needs, ICCHP'10*, pp. 263–270, Springer-Verlag, Berlin, Heidelberg (2010).
26. C. Juenemann, A. Corbin, and J. Li, "Robust door detection," Final Project, Course EE368, Stanford Electrical Engineering Department, Stanford, California (2010).
27. J. Majumdar et al., "Intelligent vision system for door sensing mobile robot," *IAES Int. J. Rob. Autom.* **1**(4), 190–202 (2012).
28. R. Muñoz Salinas, E. Aguirre, and M. García-Silvente, "Detection of doors using a genetic visual fuzzy system for mobile robots," *Auton. Rob.* **21**(2), 123–141 (2006).
29. R. G. von Gioi et al., "LSD: a fast line segment detector with a false detection control," *IEEE Trans. Pattern Anal. Mach. Intell.* **32**(4), 722–732 (2010).
30. N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst. Man Cybern.* **9**(1), 62–66 (1979).
31. C.-F. Chien, Y.-C. Cheng, and T.-T. Lin, "Robust ellipse detection based on hierarchical image pyramid and Hough transform," *J. Opt. Soc. Am. A* **28**, 581–589 (2011).
32. Y. Xie and Q. Ji, "A new efficient ellipse detection method," in *Proc. 16th Int. Conf. on Pattern Recognition*, Vol. 2, pp. 957–960, IEEE, New York (2002).
33. D. K. Prasad, M. K. Leung, and C. Quek, "Ellifit: an unconstrained, non-iterative, least squares based geometric ellipse fitting method," *Pattern Recognit.* **46**(5), 1449–1465 (2013).

**Igor Jovančević** received his bachelor's degree in computer science from the Faculty of Natural Sciences and Mathematics Podgorica, Montenegro, in 2008. He graduated from the European Master VIBOT in computer vision and robotics in June 2013. Currently, he is a second year PhD student in the Université de Toulouse, Mines Albi, Institut Clément Ader. He is working on the image and 3-D point cloud-based inspection of an aircraft within the AIR-COBOT project.

**Stanislas Larnier** obtained his thesis degree in applied mathematics from the Université Paul Sabatier, Toulouse, France, in 2011. Since then, he has been working on image and video processing for Inria,



then LAAS-CNRS. His field of research is scene analysis in difficult conditions. As a contract researcher, he is working on the detection of items on an aircraft for navigation and inspection within the AIR-COBOT project.

**Jean-José Orteu** received his PhD in computer vision in 1991. Currently, he is the deputy director of the ICA-Albi Research Center (70 people) located within the Ecole des Mines d'Albi in France. He is also the head of the Institut Clément Aders "Metrology, Identification, Control and Monitoring" group (30 people). His main interest topics are computer vision and automatic control, and he is now more specifically involved in the application of computer vision to 3-D

metrology, photomechanics, process monitoring, and nondestructive testing.

**Thierry Sentenac** is an assistant professor in the Ecole des Mines d'Albi (Mines Albi, France), and he carries out his research work at the Institut Clément Ader (ICA) Laboratory (250 people) and in the Laboratory for Analysis and Architecture of Systems (LAAS) (600 people). He is the head of program master in advanced materials and mechanical engineering, and he is also head of the "Innovative Optical Methods for Dimensional and Thermal Metrology" research group (15 people). His main interest topics are computer vision, robotics, and full-field thermal measurements.