Review

Survey on the View Planning Problem for Reverse Engineering and Automated Control Applications ☆

Manon Peuzin-Jubert [a,b,c,*], Arnaud Polette [b], Dominique Nozais [c], Jean-Luc Mari [a], Jean-Philippe Pernot [b]

[a] Aix Marseille Univ, CNRS, LIS, Marseille, France
[b] Arts et Métiers Institute of Technology, LISPEN, HESAM Université, F-13617 Aix-en-Provence, France
[c] Innovative - Manufacturing and Control, I-MC, Aix-en-Provence, France

ARTICLE INFO

Article history:
Received 8 September 2020
Received in revised form 6 May 2021
Accepted 12 July 2021

Keywords:
Point cloud
View planning
Optical sensor
Measurement
3D reconstruction
Object inspection

ABSTRACT

At present, optical sensors are being widely used to realize high quality control or reverse engineering of products, systems, buildings, environments or human bodies. Although the intrinsic characteristics of such breakthrough technologies may vary, ensuring complete acquisition relies on the definition of the optimal acquisition planning. To this end, the view planning problem (VPP) must be solved to automatically determine the optimal positions and/or trajectories of the acquisition devices to fully cover the part to be digitized. Such an automatization of the entire acquisition process is of considerably interest in the context of Industry 4.0. The aim of this paper is to review the state of the art works addressing the view planning problem and to identify the future challenges and possible research directions. First, the paper introduces a set of criteria to analyze the available methods, grouped into several macrocategories. The categories are presented and formalized to clearly understand the backbone and similarities of the grouped methods. Second, the paper describes and characterizes the available methods, based on their analysis according to the adopted criteria. The results of this extensive analysis clearly highlight the open issues and future challenges.

1. Introduction

The need to reconstruct high-resolution 3D virtual models of objects, products and systems has become mainstream in many industrial applications. The ability to reverse engineer and specifically to reconstruct or update 3D models of machined parts, buildings, historical monuments, lands and human bodies is of considerable interest in the context of Industry 4.0 as well as in building information modeling (BIM) and cultural heritage applications. With the emergence of increasingly accurate and convenient acquisition means (e.g., optical sensors, robot arms, drones), the requirement of 3D reconstruction has increased significantly in several domains, especially in manufacturing control and building surveillance applications. In recent years, the accuracy of laser scanners and fringe projection acquisition devices has been considerably enhanced, thereby making it possible to respond to an industrial demand for automated control. At present, inspection is often conducted in delayed time and in a location different from the manufacturing point. This configuration increases the cycle time and the final cost of a part. In 2018, Zhang proposed a state of the art review of the various means of measurement based on structured light [1] and discussed the advantages and drawbacks of this type of technology. One of the key steps driving the automatic reconstruction of 3D objects is the planning of views, namely, the view planning problem (VPP), aimed at automatically determining the optimal positions and/or trajectories of the acquisition means to fully cover the part under consideration.

In recent years, this problem has been studied to realize the reconstruction of postmanufactured mechanical parts by using optical sensors. In the current industrial inspection methods, probes mounted on coordinate measuring machines (CMM) are used, for high-accuracy measuring. However, in this type of inspection, the amount of time spent is considerably high compared to the number of points acquired. In contrast, optical sensors can be used to collect several thousand points in a few seconds. By installing such sensors on robots or other means of movement, multiple views of the object to be inspected can be acquired. In such a scenario, the challenge is to position and orient the sensor in space to scan the part efficiently, i.e. to cover the part as much as possible while minimizing the number of scans.

☆ This paper has been recommended for acceptance by Stefannie Hahmann.

* Corresponding author at: Aix Marseille Univ, CNRS, LIS, Marseille, France.
E-mail addresses: manon.jubert@i-mc.fr (M. Peuzin-Jubert), Arnaud.Polette@ensam.eu (A. Polette), dominique.nozais@i-mc.fr (D. Nozais), jean-luc.mari@univ-amu.fr (J. Mari), Jean-Philippe.Pernot@ensam.eu (J. Pernot).

Although view planning methods are applicable to many domains, this paper focuses on methods to realize view planning to reconstruct objects by using optical sensors. Such methods can facilitate the development of Industry 4.0. By automatically generating a view plan using an optical sensor, machined parts on machine tools can be scanned and controlled in real time without human intervention, thereby allowing the automatic adjustment of machine parameters and tool paths during machining, directly through a supervisor. At present, in general, after a part is machined to completion, it is inspected on a different machine, usually a coordinate measuring machine (CMM), and the machining is restarted with new parameters if the quality is not satisfactory. This process is repeated until the part specifications are specified. By automatically planning the acquisition views and obtaining a point cloud, controls can be automated, scrap generation can be reduced, and the product quality and competitiveness can be enhanced.

In 2003, a state-of-the art review of the existing methods was provided by Scott et al. [2]. Therefore, this survey paper is focused on the methods developed *a posteriori*. In 2020, Zeng et al. also provided a survey on the view planning problem [3]. Nevertheless this state-of-the-art only focuses on next-best-view methods and it sorts methods by type of application. The contribution of this work is threefold: (i) A set of criteria is defined to compare and classify the existing techniques; (ii) the existing techniques are extensively analyzed and systematically characterized considering the identified criteria; (iii) the open questions and future challenges are highlighted.

The remaining paper is organized as follows. Section 2 presents the issues related to the view planning problem and explains the criteria adopted to classify the methods. Section 3 and Section 4 introduce the methods that use the *a priori* knowledge of the object to be digitized and those in which no knowledge of the object is considered, respectively. The final section presents the concluding remarks with a comparative study of the methods classified according to the identified criteria.

2. Classification of the approaches and definition of the adopted comparison criteria

This section introduces the view planning problem, along with the different categories of and criteria used to characterize the different solution approaches.

2.1. View planning problem and classification of the resolution methods

The view planning problem (VPP) can be handled in two different ways: either with or without knowledge of the object to be digitized. Generally, methods that exploit the knowledge of the object to be scanned employ a CAD model or simply a mesh as the input. In contrast, the methods that do not exploit such knowledge usually start with an initial position and determine the next positions in real time.

Solving the VPP involves determining a minimum number of views that should be used to reconstruct the part in 3D. A classical method involves generalizing the VPP in the form of a more typical problem, namely, the set covering problem (SCP). This problem can be formulated as follows: If P is a set of elements $\{1, 2, 3, \ldots, n\}$, and Ps is a list of subsets whose union is equal to P, solving the SCP is equivalent to finding the smallest list of subsets in Ps such that the union of this list is equal to P. In 2001, Scott et al. [4] highlighted an approach to transpose the VPP into the SCP. If the outer skin of a 3D object is divided into several *patches*, and if one scan represents an overlap of a set of these *patches*, the goal is to overlap a maximum of *patches* with the minimum number of scans. The SCP is a classical optimization problem, and in 1972, Karp [5] demonstrated that among 21 other problems, the SCP problem is NP complete. In other words, no solution to this problem can be realized in polynomial time. Consequently, the existing methods can be classified in three categories:

- Methods based on the SCP principle that use optimization algorithms to solve the problem in a reasonable time. In this paper, these methods belong to the category "set covering problem transposition", as described in Section 3.1.
- Methods that do not seek to minimize the number of scans at all costs when an input model is used. These methods do not need to solve the classical optimization problem as the SCP is not considered in this case. Such methods are grouped in the "covering optimization" category described in Section 3.2.
- Methods solving the VPP in the absence of an input model, for instance, in *reverse engineering* applications or for area exploration applications. Here, the objective is to scan a complete object without any prior knowledge of its shape. In this context, an iterative process is realized to search for the *next optimal scan* at each new iteration, until a stop criterion is reached. These methods belong to the "search based" category described in Section 4.

Table 1
Symbols used to characterize the methods with respect to the adopted criteria.

Symbol	Meaning
✓	Criterion fully addressed by the method
∼	Criterion considered in the method but in a partial or unexplained manner
Empty	Criterion not addressed by the method
?	Information not available or unclear
NA	Criterion not applicable to the method

2.2. Adopted criteria and scoring system

The algorithms developed to solve the VPP can be characterized through several criteria. Scott et al. [2] defined a number of criteria in their paper, some of which, identified with a *, are used in this survey paper. Each criterion is identified by a letter of the alphabet to facilitate its referencing. Among the adopted criteria, several approaches are specific to the computer graphics domain, even though the considered methods are not only related to this domain. To examine these criteria, 5 types of symbols are used, as described in Table 1. In this article, the methods are evaluated with respect to each of these criteria. For instance, (a,✓) and (a,∼) characterize methods that fully or partially address a certain criterion, respectively. The evaluation techniques for these methods are summarized at the end of the document (see Table 3).

2.3. Algorithmic criteria

The following criteria refer to the algorithmic characteristics of the developed methods to solve the VPP.

2.3.1. General criteria

Independence to the sensor types (a).* The viewpoints must be generalized to ensure that they can be configured regardless of the type of sensor. The algorithm must be generalized to ensure that all viewpoint configurations can be handled regardless of the sensor technology.

Invariance to the object size (b). The algorithm must be able to consider objects of all sizes. The number of viewpoints must therefore be set considering the sizes of the sensor field and object to be scanned.

Types of support and installation constraints (c). The scanning strategy must take into account the constraints related to the parts and sensor positions, and the algorithm must generate scan configurations satisfying the multiple constraints.

Treatment of constraints (d). The algorithm must be as generic as possible to handle multiple types of constraints in a simple manner. The addition or removal of the constraints must be reasonably simple.

2.3.2. Algorithm performance characteristics

Self terminating (e).* The algorithm must be able to determine when the objective is attained and stop the process autonomously (without human intervention).

Scan number minimization (f). The algorithm must seek to minimize the number of viewpoints used in the scan plan.

Cover maximization (g). The algorithm must seek to maximize the covering of the surface to be scanned.

Time inspection minimization (h). The algorithm must determine a scanning strategy that minimizes the inspection time. For example, the algorithm should compute an order between each of the scans to minimize the dead times at the time of inspection.

2.3.3. Object constraints

Type of entity selection (i). The scan plan is built considering a CAD model. The algorithm must be able to identify the parts of the object associated with the scan plan and create scan overlaps in these particular areas. This criterion is important according to the industry needs, in order to increase the efficiency of the machining and geometric tolerance control processes. This criterion is deeply related to the time inspection minimization criterion (h).

A priori knowledge of the object (j).* The algorithm must have a minimum knowledge of the object to be scanned.

Overlap between scans (k).* The overlaps between the scans obtained from different viewpoints can be used to obtain correspondences between the scans. This aspect is essential to register the point clouds and reconstruct 3D objects.

Analysis of the degrees of freedom for registration (l). The point cloud registration often relies on the *iterative closest point* (ICP) algorithm. The principle is to identify the alignment of the point clouds that can minimize the distance between the clouds. The accuracy generally depends on the shape of the point clouds, which depends on the degrees of freedom that characterize the contact between the point clouds. Therefore, it may be relevant to analyze the shapes and degrees of freedom of the surfaces to be aligned to determine whether a suitable alignment of the scans can be performed.

2.3.4. Sensor constraints

Occlusion treatment (m).* The algorithm must take into account the geometry and shape of the part to detect the surface occlusions and consider them when generating a scan plan covering the whole surface.

Sensor quality measure (n). The algorithm must consider the characteristics and limitations of the acquisition technology to minimize disruptions and errors, for example, the inclination or distance of the sensor from the surface.

Collision detection (o).* When generating the scan plan, the algorithm must avoid collisions between the part, sensor, support and environment.

2.3.5. Validation conditions (p)

To ensure consistency with the industrial conditions, the results must be validated considering real industrial parts in a real environment with potential obstacles.

2.3.6. Type of discretization

The considered criteria help approximate the types of geometric representations that can be manipulated, which helps clarify the associated advantages and disadvantages. The methods examined in this paper involve five types of discretization processes, which can be used as an input of the methods or as intermediate representations adopted to satisfy the algorithmic needs (Fig. 1).

Mesh (q). In the context of the VPP, a mesh is a discrete representation of an object's outer surface and is often used to accelerate computation. The mesh is composed of points, edges and faces. Depending on the type of mesh used, it can be organized in different ways. For example, the edges can be oriented, and the mesh can have a certain direction of travel. Triangular meshes are often used for display purposes; however, they can also be used for processing when subdivided homogeneously.

Voxel grid (r). The voxel grid is used to represent an object in a simple manner and to represent the space areas discretized by a three dimensional grid. Depending on the type of structure used, the voxels can be labeled in a binary manner or according to several labels following certain criteria. This structure is particularly effective to perform a neighborhood search.

Parametric surfaces (NURBS) (s). Parametric surfaces allow the representation of complex shapes. For instance, Non-Uniform Rational B-Spline (NURBS) [6] surfaces allow a complex object to be represented in a simple manner by using control points, knot sequences and weights. It is generally used for the analytical description of a surface.

Point cloud (t). Point clouds are often derived from a data acquisition, and they allow a real object to be represented as a set of 3D points.

B-Rep (Boundary representation) (u). B-Rep representations [7] are widely used in the industrial domain to describe solids such as CAD models. A B-Rep object is composed of several elements that constitute the object's skin. It covers aspects related to the geometry and the topology of the skin. Faces of a model can be represented by means of parametric surfaces (NURBS, B-Spline, etc.) and connected to each other by so called wires composed of edges and vertices.

2.4. Technological criteria

A part of the considered methods work only with one type of sensor or are limited to specific applications. Consequently, the approaches must be characterized with respect to a set of technological criteria. Such criteria allow users to choose the appropriate method according to their specific constraints. The characteristics of the existing approaches with respect to these criteria are summarized at the end of the paper (see Table 2).

2.4.1. Acquisition technology

The digitization phase can be realized using various acquisition technologies, which can be classified in three categories.

Laser scanners. Laser scanners usually involve a camera. A ray is projected onto the surface to be scanned, and the scanner is moved to obtain the complete digitization of the surface (Fig. 2).

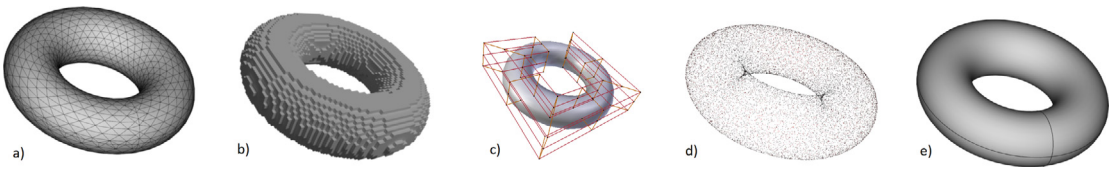

Fig. 1. Various tore representations: Mesh (a), Voxel Grid (b), Parametric Surface (c), Point Cloud (d), B-Rep (e).

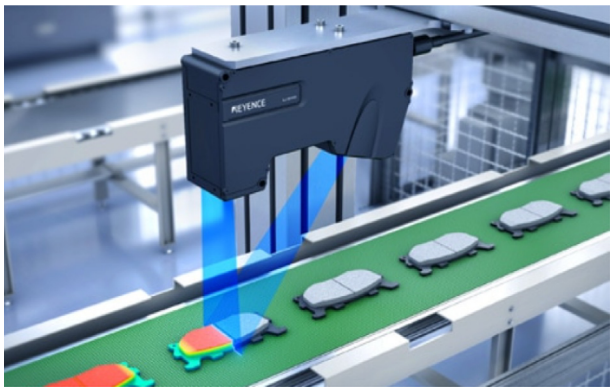

Fig. 2. Example of Keyence© brand laser scanner.

Structured light sensors. Structured light sensors generally involve one or two cameras. The sensors project a set of light patterns onto the object to be scanned and observe the way that the patterns deform, to reconstruct the surface within a field specific to each sensor. The sensor is fixed during the acquisition (Fig. 3).

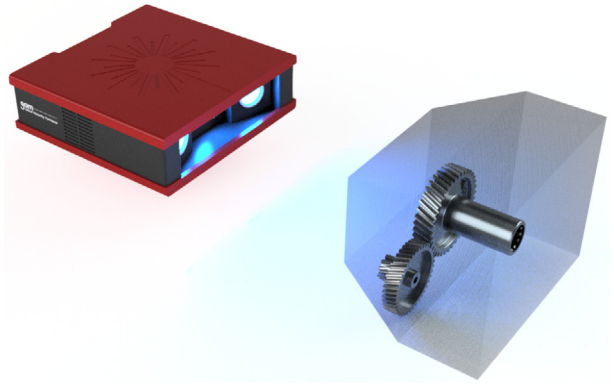

Fig. 3. Example of GOM© brand structured light sensor.

Others sensors. Several other types of acquisition devices, using cameras, wireless sensors, probes, etc. have been reported in the literature.

2.4.2. Type of support

To establish a scan plan that allows the entire part to be scanned, the acquisition system must be able to move around it. Different strategies can be used depending on the available means. For instance, the part can move according to its support, the acquisition device can move around the part, or a combination of both strategies can be employed.

Turning table. A turntable can be used as a support for the part to be scanned, if the part is not excessively large. The table can generally rotate around an axis, or even two or three axes in certain cases, to allow data acquisition from several viewpoints (Fig. 4).

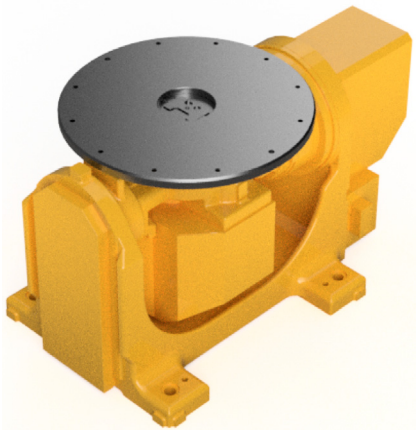

Fig. 4. Example of FANUC© brand motorized turning table.

Robotic arm. A robotic arm is often used to approach areas that cannot be easily accessed. The sensor is fixed at the end of the arm. Depending on the size of the arm, the sensor can move around and orient itself in different configurations with respect to the part. The robotic arms reported in the literature can move in six to eight axes (Fig. 5).

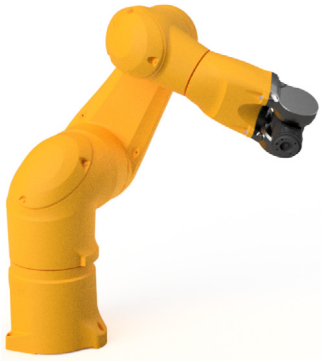

Fig. 5. Example of Staübli© brand robotic arm.

CMM. Coordinate measuring machines are traditionally used to measure a machined part by using probes. The machine consists of a table and a gantry that moves in three axes, with a measurement head installed at the end. Although such machines have a high accuracy, the measurement process is slow as the points are acquired individually (Fig. 6).

UAV. Several of the considered methods employ unmanned aerial vehicles (UAVs). UAVs are small remotely controlled aerial vehicles equipped with acquisition devices such as LIDAR, and they are used to reconstruct large objects, such as monuments or land, whose parts cannot be reached otherwise (Fig. 7).

Mobile robot, AGV. Mobile robot are able to move in an environment or to carry heavy objects (8). They are useful to explore dangerous environments for humans (when there are radiations for instance). In the process of view planning they can be used to

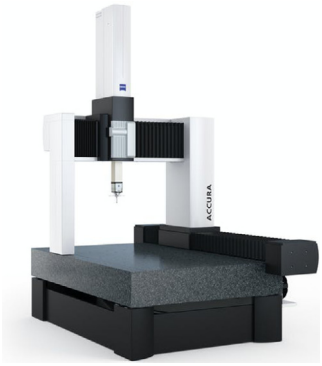

Fig. 6. Example of ZEISS© brand CMM.

Fig. 7. Example of DJI© brand UAV.

Fig. 8. Example of IMR System© brand mobile robot.

move around an object and to scan with a sensor attached to it, or to carry a robotic arm equipped with a sensor.

2.4.3. Application type

The different types of VPP applications can be classified into four categories in order to choose properly the tools and the data processing. The classification is important because search-based methods are more suitable for the digitization of unknown objects and the model-based methods are more appropriate for the digitization of known objects. Also the size of the object could influence the processing time. Here, objects are considered "small" when they fit into twice the length of the scanner field area, in the opposite case they are considered as "large".

Digitization of small unknown objects. Such applications involve inspecting mechanical parts or reverse engineering realistic 3D objects to feed databases for 3D model learning or 3D simulations.

Digitization of large unknown objects. Such applications involve the reconstruction of complex external structures such as buildings, historical monuments, statues of even large outdoor scenes and environments.

Digitization of small known objects. The reconstruction of small objects, whose shape is known *a priori*, is a key focus area in this review. Such applications pertain to the 3D reconstruction of an object to enable its inspection at the end of the manufacturing process.

Digitization of large known objects. As in the case of small known objects, such applications involve the inspection of large parts and the reconstruction of historical monuments, towers or buildings, whose geometry is known *a priori*.

3. Approaches based on *a priori* knowledge

This section describes methods that exploit the *a priori* knowledge of the object to be digitized (j✓), in contrast to *search based* approaches, in which the shape of the object is not required to be known. In this case, one of the inputs of the algorithm is a 3D representation of the object to be scanned (e.g. mesh, point cloud, volume, CAD model).

In particular, two types of algorithms exist: methods that seek to solve the SCP, and methods that seek not to minimize the number of scans but maximize the coverage of the object.

3.1. Approaches based on solving the SCP

According to the definition presented in Section 2.1, the elements of set P correspond to the parts of the surface to be scanned, known as patches. The list of subsets Ps is represented by a list of the viewpoints. Each viewpoint pv is defined by coordinates pos to compute the set of patches p_i visible from this point.

To solve the SCP, the existing methods involve five main steps (Fig. 9). The first step involves the segmentation of the surface into patches. The object model to reconstruct is sampled into n patches of p_i surfaces in the set P. The second step involves sampling the space into viewpoints. A list Ps of viewpoints pv is sampled. A viewpoint pv is generally described with a vector pos that represents a (x, y, z) position and an orientation (rx, ry, rz), and a vector w_{pv} that contains the patches p_i of the object surface belonging to P visible from pv. At the beginning of the method, the vector w_{pv} is empty. The third step involves an evaluation of the viewpoints. At this step the vector w_{pv} of the viewpoints pv are filled. The fourth step involves the resolution of the optimization problem. The number of pv which covers the maximum of surface patches p_i is minimized. Finally, the fifth step involves identifying the criteria used to optimize the objective function.

All the methods in this section follow the same steps presented in the formalization of the view planning problem resolution in Fig. 9.

3.1.1. Surface segmentation methods

Among the various surface segmentation methods, the most common approach is to use a mesh of the object and consider a face of this mesh as a patch p_i. The meshes can involve different levels of details, and often, a simplified mesh is used to approximately represent the shape of the object. Loriot in 2009 [8], Scott et al. in 2002 [9,10] and Mahmud et al. in 2011 [11] used this method because it can reduce the complexity of the algorithm,

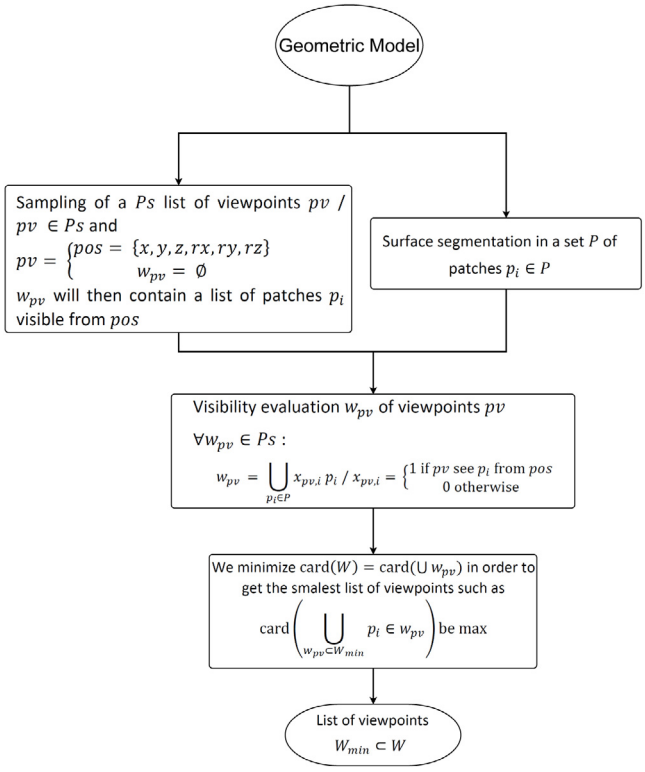

Fig. 9. Formalization of the 5 main steps involved in the methods to solve the SCP.

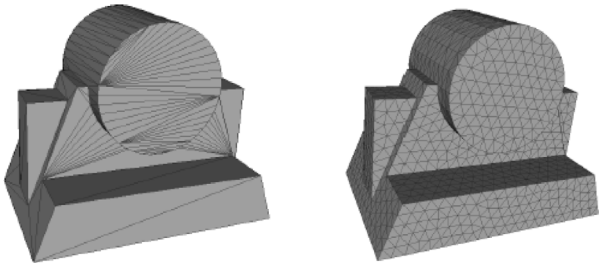

Fig. 10. Original mesh (left) and mesh subdivided using the bubble mesh algorithm [14,18] (right).

and therefore, the calculation time (q ✓). However, if the object is complex, the self occlusions of the surface may be lost when using this approach.

In 2009, Scott [12] decided to use the decimated mesh of a model (q ✓). In this approach, the simplification level is determined experimentally for each scan plan. Thus, the method is not automatic, and a manual preprocessing must be performed. Similar to the condition in the aforementioned approaches, mesh simplification results in the removal of occlusions, and thus, potentially important features. In all the methods proposed by Jing et al. [13–16] and Mohammadijaki et al. [17], the employed mesh is subdivided to obtain a homogeneous mesh over the entire surface (q ✓). Jing et al. used the bubble mesh algorithm [18] (this method generates a uniform triangular mesh which preserves the original shape), the process flow of which is illustrated in Fig. 10. In this approach, the faces of the mesh are homogeneous sized over almost the entire surface, even in the case of a flat surface. Consequently, the sizes of the patches (i.e. segmentation) are similar.

Another way to segment a surface into patches is to position the object in a grid of voxels. In 2005, Martins et al. [19] used this method (r ✓). This method allows the realization of a structure that can be used to detect collisions between the object and scanner, for example. Hepp et al. [20] also used this method (r,o ✓).

In 2011, Krause et al. [21] used a grid to segment the surface, although a simple 2D grid was used instead of voxels, because the surface to be covered was planar (r ∼).

3.1.2. Viewpoint sampling methods

In these methods, a key aspect is to generate the viewpoints. If the determined viewpoints do not cover the surface suitably, the VPP cannot be solved.

Orientation with a normal surface A widely used approach involves generating a set of viewpoints per patch, with the viewpoints aligned against the normal of the patch. Scott [9,10], Martins et al. [19] and Mahmud [11] used this method in their respective research. In 2011, Krause et al. [21] generated a set of positions for each patch, although no orientation was generated, as it was not required in their algorithm. Although this sampling method is simple, it depends considerably on the surface segmentation. If the mesh or its segmentation are excessively coarse, the self occlusions cannot be considered, because the orientations are defined according to the normal of each patch.

In 2009, Scott et al. [12] and Loriot [8] generated viewpoints along the normal to each point on the surface. Subsequently, a filter was applied to all the viewpoints to reduce the complexity. Nevertheless, similar to the limitation of the aforementioned approach, this type of sampling was also considerably dependent on the segmentation step.

Random sample in a volume space Jing et al. [13] randomly sampled the positions of the viewpoints within a specific volume. This volume involved two volumes. Specifically, the first and second volumes were obtained by dilating the surface with the minimum and maximum distances of the camera field, respectively. The final volume was the intersection between these two volumes. To compute the orientations, each object was considered to have a force of attraction. Thus, for each viewpoint, the forces of the closest patches were summed, and the resulting normalized 3D force corresponded to the direction of the viewpoint.

In 2017, Jing [14,15] used the same method, albeit instead of the volume, the viewpoints were sampled on the *medial object* of the previous volume. Subsequently, the viewpoints were preselected according to certain criteria to reduce the computation time. This method could be used to sample the viewpoints to enable the visualization of the surface. However, since the positions of the viewpoints were sampled randomly, and the orientations were computed according to the force of attraction of the surface, certain parts of the surface were likely never observed from the sampled viewpoints.

Graph method Hepp et al. [20] created a graph of candidate viewpoints. The nodes of the graph corresponded to the viewpoints, and the voxels observed from these viewpoints were recorded. The edges corresponded to a collision less path between the two viewpoints. The principle of this method was to reduce the graph to a minimum graph to cover the surface to be scanned. The candidate viewpoints were generated iteratively. At each created viewpoint, 6 new viewpoints were added with an offset in the directions $-x$, x, $-y$, y, $-z$ and z. The viewpoints were retained if they were not too close to the already generated viewpoints, and if they were located in a free access space. The orientations were determined through random sampling, while preferring the points located close to the regions of interest. The advantage of this method was that the generated viewpoints were located in an accessible space free of collisions.

Sampling in robot kinematic space Jing et al. [16] sampled the viewpoints on an ellipsoidal hemisphere containing the object, and for each viewpoint, a robot configuration was defined in the robot kinematic space (c ✓). The viewpoints for which no configuration existed, or which involved a collision with the environment, were directly removed from the set. The advantage of this method is that the viewpoints can be directly filtered at the time of their generation, based on the occurrence of a collision and robot accessibility criteria (o, ✓). However, this aspect can be a disadvantage, as the method is constrained to a specific workspace for a robot arm whose accessibility can be directly tested. This phenomenon does not occur in the case of other types of media. Moreover, the size of the objects is constrained by the maximum extension of the robot arm (b ∼).

Mohammadikaji et al. [17] did not simply sample a finite number of viewpoints as the first step. In each iteration of the optimization algorithm, a new random sampling of the viewpoints was performed within the search space predefined by the sensor and model support. This working method is similar to that of search based methods that search for the next optimal scan at each iteration. The advantage of this method is that all the viewpoints are renewed at each iteration, which theoretically increases the possibility of finding a better viewpoint. However, the efficiency of this approach is low, since unlike in the case of other methods that sample and evaluate viewpoints only once, the process is performed each time a new viewpoint is calculated. In the approach used by Jing et al. [16], the viewpoints are sampled and optimized in a space defined by the support (c ✓). The method is thus valid only for the support for which it is defined, and the size of the object is also constrained by the size of the positioning space (b ∼).

Apart from the methods of Jing et al. [16] and Mohammadikaji et al. [17], all the methods presented in this section can be considered to be generalized for any object size (b ✓).

According to the definition of the viewpoints, a method may be independent to the measurement technique. Most of the considered methods use generalized viewpoints. Specifically, the methods proposed by Hepp et al. [20], Jing et al. [13–15], Mahmud et al. [11], Scoot et al. [10,12] and Loriot [8] define a viewpoint in terms of a position and a second parameter, which can be an orientation, or a set of parameters, such as an orientation and a scan direction, such as for laser scanners (a ✓). However, Martin et al. [19] defined the viewpoint in terms of only the orientation and calculated a path for the laser scanner. No position in space was defined, and only the distance between the surface and scanner was fixed according to the sensor characteristics. The viewpoints were not generalized, although they could be easily generalized by dividing the scan path into several parts sized as the sensor field, if the sensor was not a laser scanner (a ∼). Krause et al. [21] defined a point as a simple coordinate in space since the adopted method and positioning means did not require any additional information. Jing et al. [16] and Mohammadikaji et al. [17] positioned viewpoints in the space of the positioning mean. Consequently, the viewpoints were specific to the type of mean and could not be generalized to other acquisition devices.

3.1.3 Visibility evaluation methods

The methods to evaluate the visibility of a surface from a viewpoint usually exploit a binary measurability matrix for each patch of the surface. These matrices were derived by Tarbox et al. [22]. The algorithms used by Scott et al. [9,10,12], Loriot [8], Jing et al. [13–16] and Martins et al. [19] exploit the measurability matrices to evaluate the visibility of each patch from different viewpoints. An example of such a matrix is shown in Fig. 11. This matrix can be used to efficiently determine the surface patches visible from different viewpoints (m ✓). The quality of the matrix depends on both the quality of the segmentation of the surface and sampling of the viewpoints. However, the computation of these matrices is time consuming and must be performed during the preprocessing of the algorithm. Another way to evaluate the visibility of each patch is to create a visibility cone for each patch, as performed by Mahmud et al. [11]. These cones determine a set of orientations from which the patch is visible (m ✓). Thus, the patches with a nonempty intersection of their cones can potentially be simultaneously visible. Nevertheless, these cones cannot be used to simultaneously evaluate the visibility of the patches and scan them with a well defined field, such as in the case of structured light sensors. In the case of laser scanners, an orientation must be associated with a scan path, and a simple orientation may also be employed.

In contrast, in the approaches of Krause et al. [21], Hepp et al. [20] and Mohammadikaji et al. [17], a visibility function is used for each viewpoint. In other words, each viewpoint is evaluated according to a predefined function with certain criteria defined specific to each method.

3.1.4 Objective function criteria

The approaches based on solving the SCP maximize the coverage of the surface while minimizing the number of viewpoints (f ✓,g ✓). These algorithms minimize or maximize a so called cost function or objective function, respectively. The definition of this function is decisive for the result to be obtained, and it can be characterized using several criteria.

The first criterion corresponds to the surface coverage. Scott et al. [9,10,12], Loriot [8], Krause et al. [21], Jing et al. [13–16], Hepp et al. [20], Martins et al. [19] and Mohammadikaji et al. [17] integrated this criterion when defining their objective functions. Mohammadikaji et al. [17] considered the selected CAD model from the perspective of the specific region to be covered (i ✓), and therefore did not intend to achieve a complete coverage of the model at all costs.

Nevertheless, in the VPP, the coverage is a necessary but insufficient criterion, and other aspects must be considered. In addition to optimizing surface coverage, Jing et al. [13–15], Hepp et al. [20], Martins et al. [19] and Mahmud et al. [11] attempted to optimize the scanner's direction based on the surface normal (n-checkmark). This principle helped enhance the quality of the point cloud. In 1999, Prieto [23] demonstrated that the angle between the surface normal and sensor orientation should not exceed 35°, because beyond this angle, the resulting point cloud is extremely noisy and cannot obtain accurate measurements.

To evaluate the coverage of the surface from a viewpoint, Jing et al. [13,14,16] and Hepp et al. [20] considered the effects of occlusions. Several methods, such as those of Loriot [8] or Scott [9], simplify the model of the object and attempt to scan only the simplified model. Depending on the simplification rate, certain parts of the real surface may not be reached because of the occlusions not considered.

Another key criterion to be optimized is the overlap between scans. In computer graphics, to align two point clouds, the registration algorithms rely on identifying a common part between the two point clouds. When considering a complete scan plan, because several measurements must be performed, several alignments must be realized to reconstruct the point cloud of the complete part. Consequently, the criterion pertaining to the fact that each viewpoint in a scan plan must have at least one part in common with another viewpoint must be optimized. Scott et al. [9,12], Krause et al. [21], Jing et al. [15] and Hepp et al. [20] used this criterion in their objective functions (k ✓).

Moreover, no collisions should occur between the acquisition means, object to be scanned and obstacles in the environment.

	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8	v_9	v_{10}	v_{11}	v_{12}	v_{13}	v_{14}	v_{15}	v_{16}	v_{17}	v_{18}	v_{19}	v_{20}
s_1	1	1	1	1	1	0	0	1	1	1	0	0	1	1	0	0	0	0	0	0
s_2	1	1	1	1	1	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0
s_3	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0	1
s_4	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	1	0	0	0	1
s_5	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
s_6	0	0	1	1	1	1	1	1	0	0	1	0	0	0	0	1	0	0	1	1
s_7	0	0	0	1	1	1	1	1	0	0	1	0	0	0	0	0	0	0	1	1
s_8	1	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0
s_9	1	0	0	0	0	0	0	1	1	1	1	1	1	0	1	0	0	0	0	0
s_{10}	1	1	0	0	0	0	0	0	1	1	0	1	1	1	1	0	0	0	0	0
s_{11}	0	0	0	0	0	0	1	0	0	0	1	1	0	0	1	0	0	1	1	1
s_{12}	0	0	0	0	0	0	0	0	1	1	1	1	1	0	1	0	0	1	1	0
s_{13}	1	1	0	0	0	0	0	0	1	1	0	1	1	1	1	0	1	1	0	0
s_{14}	1	1	0	0	0	0	0	0	0	1	0	0	1	1	1	0	1	1	0	0
s_{15}	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0	1	1	1	0
s_{16}	0	1	1	1	0	1	0	0	0	0	0	0	0	0	0	1	1	0	1	1
s_{17}	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0
s_{18}	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	1	1	1	0
s_{19}	0	0	0	0	0	1	1	0	0	0	1	1	0	0	1	1	1	1	1	1
s_{20}	0	0	1	1	0	1	1	0	0	0	1	0	0	0	0	1	0	0	1	1

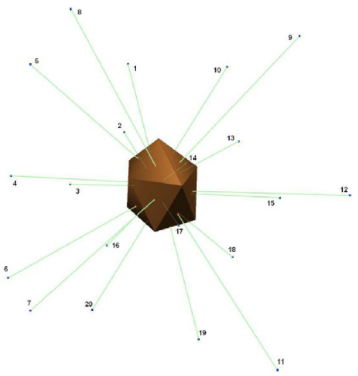

Fig. 11. Matrix of measurability (left), where v_i and s_i denote the viewpoints and patches of the surface (right), respectively [8].

Scott et al. [12], Hepp et al. [20], Jing et al. [16] and Martins et al. [19] integrated the collision criterion directly into the objective function to avoid selecting viewpoints that would collide with a surrounding element (o ✓).

In the industry, the scan plan is used to scan parts to be inspected and controlled, to reduce the inspection time. One of the criteria used by Hepp et al. [20] and Jing et al. [16] is to optimize the inspection time (h ✓). To this end, the considered criterion pertains to the distance between a newly identified viewpoint and previously selected viewpoints. The type of distance can vary according to the means of displacement of the sensor (e.g. robot or drone).

In general, by using an objective function, constraints can be added or removed as desired (d ✓), as long as the constraints can be evaluated in the created virtual environment.

3.1.5 SCP resolution methods

The set covering problem is an NP complete problem, and thus, the optimal solution cannot be found in a reasonable time. To solve this problem, many optimization algorithms try to approximate the solution in a finite time. To this end, one of the most widely used algorithms is the greedy algorithm [24], which is an iterative algorithm that locally chooses the optimal solution at each iteration. A disadvantage of this method is its tendency to fall into local optimums. Nevertheless, this approach is easy to implement and, under a relevant cost function, the approach is sufficiently efficient to be used to solve NP complete problems. Scott et al. [9,10,12], Jing et al. [13,15] and Martins et al. [19] used only the greedy algorithm to optimize the problem (e ✓). Certain other optimization methods reported in the literature drew upon the greedy method and improved it to better fit the application requirements. The approach of Hepp et al. [20] is a recursive greedy algorithm that ensures a balance between the scan optimization and optimization of the distance between the scans. Specifically, this algorithm ensures a balance between the greedy and cost benefit algorithms (e ✓).

Jing et al. [16] also proposed a greedy approach, although the Monte Carlo tree search (e ✓) was used in this case. At each iteration, starting from a starting node, a search tree was established. Subsequently, the algorithm attempted to identify the best "child" in this tree, which became the next starting node. This method, combined with an ad hoc cost function enabled the determination of a solution to not only the SCP, but the traveling salesman problem (TSP), while minimizing the inspection time.

In addition, Krause et al. [21] proposed a new greedy approach to solving the SCP, namely, the sensor placement at informative and cost effective location (SPIEL). The principle is as follows. The possible positions are decomposed, and the greedy algorithm is used to order the positions in each cluster. A string is then generated to link the positions. The first nodes of each chain is linked to form a complete graph (modular approximation graph G′) in which the edges are weighted by the communication costs. An approximate solution to the maximization problem is then found on G′ (represented by a set of edges), and the solution is extended to the graph G, thereby obtaining the shortest path to cover a maximum area (e ✓).

Moreover, to solve this optimization problem, Loriot [8] used a tool developed by Lan et al. [25], in which, the metaheuristic for randomized priority search (MetaRaPS) was employed. This solution was considered to be a general form of the greedy, COMSOAL and greedy randomized adaptive search procedure (GRASP) algorithms developed by DePuy et al. [26]. The tool helped improve the selection of a solution at each iteration of the greedy algorithm and in penalizing solutions whose search space was condensed in one location (e ✓).

The use of genetic type optimization algorithms has also been reported in the literature. Jing [14] used a random key genetic algorithm combined with a greedy algorithm to solve the coverage problem. In particular, an implementation available in MATLAB [27] was employed. The principle was to encode the information with a random key between 0 and 1 and to store the keys in the genes of a chromosome. The decoding process consisted of sorting the genes by values. The cost function was evaluated by adding the sorted genes individually to the solution until the coverage constraint was satisfied. Although genetic algorithms are generally time consuming, in this case, the initialization was close to the solution (since it was similar to the partial solution found using the greedy approach) and the problem optimization was prompt (e ✓).

Mohammadikaji et al. [17] combined the greedy optimization method with particle swarm optimization (PSO). The principle was to avoid the state of being confined to a set of positions defined at the beginning of the process. With each new iteration of the greedy algorithm, certain parameters in the search space were initialized to create an initial particle swarm, and subsequently, a particle swarm optimization process was performed to find the next optimal position (e ✓). The disadvantage of such algorithms is the low time efficiency in most cases. In the approach of Mohammadikaji et al. [17], an iteration based optimization was performed, owing to which, the process was extremely long, and the effort was comparable to that of manual computation.

3.1.6 Algorithm validation

Various applications have been considered in the literature, and numerous tools have been proposed to develop a test platform and validate the algorithms.

Scott et al. [12], Martins et al. [19], Loriot [8] used view planning to rebuild all kinds of models. To this end, Scott et al. [9,10,12] and Loriot [8] used structured light sensors, whereas Martins et al. [19] used a laser scanner mounted on a CMM.

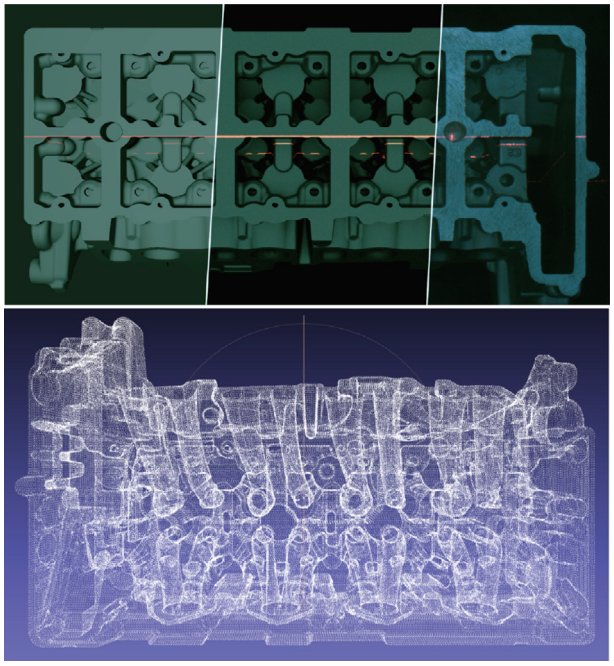

Fig. 12. Top: (left) fast rasterization based simulation to evaluate the measurement coverage; (middle) simulations with path tracing, considering the camera spectral response; (right) real camera image. Bottom: Point cloud scanned by applying 30 acquisitions optimized through the greedy planning realized using the method of Mohammadikaji et al. [17].

The inspection of mechanical parts has been extensively examined (a✓,b?), notably in the works of Mahmud et al. [11], Jing et al. [14,16] and Mohammadikaji et al. [17] (see Fig. 12). Jing et al. [14,16] used a structured light sensor mounted on a robotic arm, with the part placed on a rotary table. In contrast, Mohammadikaji et al. [17] used a laser scanner. Similar to Martins et al. [19], Mahmud et al. [11] mounted their laser scanner on a CMM.

In 2017, Jing et al. [14,15] expanded their work to building inspection and surveillance applications, as well as to reconstruct the surface of a large outdoor statue [13]. Hepp et al. [20] also proposed the application of their algorithm to reconstruct outdoor environments. The acquisition means used in these methods were cameras mounted on UAVs.

Krause et al. [21] proposed an application that was slightly different from the existing approaches. The recovery algorithm was not used for any reconstruction, but to support the placement of thermal sensors to control the temperature on the floor of a building (i NA)(p NA). Even though temperature and humidity sensors were used instead of optical sensors, the principle remained the same, since the goal was to cover an area by placing the sensors at a sufficient distance, thereby allowing efficient communication between the sensors; this aspect can be compared to the overlap between each scan of the surface.

The types of object on which a method is validated can vary widely. In many cases, classical models such as the Stanford Bunny, a mask, a gnome, a hairdryer or the model of a woman's body are used [12,16,19]. Mohammadikaji et al. [17] (p ✓) and Mahmud et al. [11] (p ∼) validated their methods by using industrial mechanical parts. In this regard, certain studies can be considered to be preliminary considering the experiments conducted. Scott et al. [10] and Jing [14] tested their method on objects with simple shapes and with little or no change in curvature. Such testing techniques cannot fully validate the approaches, considering the highly complex nature of industrial applications. Several of these validations were performed through simulation, as in the works of Jing et al. [15] and Hepp et al. [20] (i NA)(p NA). Jing et al. [13,14] validated their methods on a statue in its outdoor environment as well as on an actual building.

3.1.7 Summary of SCP resolution based methods

The methods presented in this section are based on a transposition of the VPP into a more general graph theory combinatorial optimization problem, that is, the SCP.

All the identified methods follow the same global scheme, as formalized in 5 main steps. In the first step, the surface is segmented to create a set P of patches. To this end, two types of methods can be used. The first method is to mesh the surface and consider that a face of the mesh represents a patch of the surface. This method was used by Loriot [8], Scott et al. [9,10,12], Mahmud et al. [11], Jing et al. [13–16] and Mohammadikaji et al. [17]. The second method employs a grid, in which each cell represents a patch. This approach was adopted by Martins et al. [19], Hepp et al. [20] and Krause et al. [21].

In the second step, the viewpoints are sampled to obtain a list of subsets Ps. To this end, two groups of methods can be used: The first group is based on the segmentation of the surface to determine a viewpoint for a given patch, as used by Scott et al. [9,10,12], Martins et al. [19], Mahmud et al. [11], Krause et al. [21] and Loriot [8]. The second method does not perform any surface segmentation and instead samples viewpoints randomly in a defined space, as performed by Jing et al. [13–16], Hepp et al. [20] and Mohammadikaji et al. [17].

The third step is to evaluate the visibility of the sampled viewpoints. Two types of methods can be used to conduct this step: The methods that evaluate the visibility of a patch, used by Scott et al. [9,10,12], Loriot [8], Jing et al. [13–16], Martins et al. [19] and Mahmud et al. [11], and those that evaluate viewpoints, as used by Krause et al. [21], Hepp et al. [20] and Mohammadikaji et al. [17].

The final step is to solve the minimization problem. To this end, two groups of methods can be identified. The first group employs different variants of the greedy approach and was used by Scott et al. [9,10,12], Jing et al. [13,15,16], Martins et al. [19], Hepp et al. [20], Krause et al. [21] and Loriot [8]. The second group, which was utilized by Jing [14] and Mohammadikaji et al. [17], employs metaheuristics to solve the problem.

The key difference among these approaches is the structuring of the cost function to be minimized. The result varies depending on the parameter that each function intends to minimize. The criteria used are specific to each method, and thus, the methods cannot be grouped in this scenario.

3.2 Approaches based on coverage optimization

This section attempts to classify methods in which the view planning problem and set coverage problem are not directly compared. The objective of these methods is to optimize, first, the coverage of the object to be scanned, and later, the number of scans. Specifically, these methods focus mainly on the technique to segment the surface into patches and involve four main steps. The first step is to segment the surface into patches. The object model to reconstruct is sampled with n patches of p_i that are part of the set P. The second step involves optimizing the position and orientation of the scan. Each point of view pv is individually constructed with a vector pos which contains the position (x, y, z) and the orientation (rx, ry, rz), computed in order to determine a vector w_{pv} that contains a maximum of patches p_i of the object surface. The third step consists in evaluating the visibility of patches p_i from viewpoints with the computation of the w_{pv} vector, and in also optimizing the viewpoint positions. The last

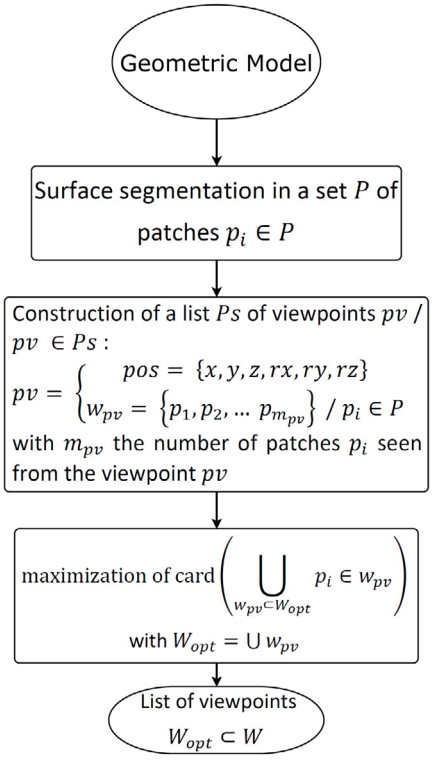

Fig. 13. Formalization of the resolution steps for methods based on coverage optimization.

step involves computing the required number of scans in order to maximize the patches p_i included in all the w_{pv} selected vector viewpoints. Fig. 13 shows how the methods based on coverage optimization can be formalized around these four main steps.

3.2.1 Surface segmentation methods

In these methods, the result is dependent primarily on the way that the surface is segmented. Consequently, several different types of segmentation methods have been developed.

Sadaoui et al. [28], Lartigue et al. [29] and Koutecky et al. [30] performed an adaptive voxelization of the model as a function of both the surface normals and size of the sensor field (r ✓). Specifically, first, a grid of voxels is created, with each voxel having the size of the sensor field. It is ensured that the angle between all the surface normals contained in each voxel is not greater than a certain threshold. Subsequently, the voxel is divided in two, and so on, iteratively. This method has certain advantages. For instance, by using a voxel grid, collisions (o ✓) can be easily detected, and all the volumes (e.g. sensor fields) can be easily represented. However, this segmentation method can turn greedy rapidly and segment the surface into extremely small voxels, for instance, in the case of a complex model with many changes in the curvatures and normals. Moreover, in this method, the normals must be calculated accurately. Sadaoui et al. [28] proposed the conduction of a "preparation" step before voxelizing the model. In this step, the CAD model is tessellated in the form of a mesh such that the faces are at most 50% the size of the sensor field (q ✓). The CAD model is decomposed into entities to be and not to be inspected (i ✓). This decomposition helps the approach focus on the scan plan and the accuracy of the measurement of the parts of the model to be measured.

In 2009, Germani et al. [31] performed the segmentation of a surface according to a sphere encompassing the object, divided into several sectors. However, this segmentation method works only if the object is convex, and because even the shape of the object is not considered, occlusion issues may occur.

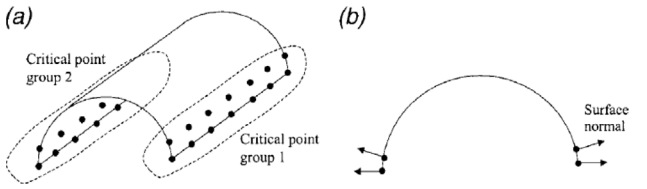

Fig. 14. Critical point groups: (a) cylindrical surface, (b) cross-sectional view [32].

Several view planning algorithms tend to segment the surface by sampling the surface as a point cloud and clustering the points according to the normals and distances. Lee et al. [32] and Son et al. [33] sampled the points along the parametric curves of the model and computed the so called critical points (t ✓, s ∼). If the angle between the normal vectors of two points was greater than a certain value, the points were identified as critical and grouped into regions (see Fig. 14).

Raffaeli et al. [34,35] proposed several methods to segment a surface. One approach was similar to the aforementioned approach, with the surface sampled in a point cloud. The points were grouped according to their normal and distance by using the k means algorithm. Another approach was to divide the surface into patches by performing the NURBS parameterization of the surface (t, s, q ✓). Each patch was defined smaller than the sensor field size to ensure a margin for overlap (k ✓). In 2013, the authors sampled the point cloud on the edges of the model rather than on the surface. These methods are reasonably effective as they consider the risk of occlusion by using normals and the size of the sensor field as well. Moreover, these approaches require the clustering part to be parameterized, in order to implement the k means technique, but the way this is performed is not fully explained.

In 2018, Phan et al. [36] used parametric surfaces to segment a surface. The 3D mesh of the object was transformed into 2D parametric surfaces, and parallel planes were generated on this surface (q ✓, s ∼). The direction of the scans depended on the length of the parametric surface. Using the same concept, Wu et al. [37] projected the points onto a 2D plane and divided the plane into a rectangle with the size of the sensor field. The method of Wu et al. [37] appears to be suitable only for objects without concavity, because the surface of the model is projected perpendicularly on the 2D plane (q ✓). Furthermore, the occlusion problems are not taken into account.

Normals are commonly used when segmenting the surface; in fact, certain approaches use normals to create "cones" of visibility. This concept is used to represent a set of sensor orientations around the normal of a point on the surface from which it is visible. Lee et al. [38], Souzani et al. [39] and Ding et al. [40] used this approach. Specifically, the points on the surface whose "cones" of visibility intersect were considered to be part of a set of points visible from the same orientation. Lee et al. [38] sampled the surface along the parametric curves of the model and visibility cones, and the so called LADs were calculated for each point (t ✓). Similarly, Souzani et al. [39] discretized their model as a grid of voxels; for each voxel containing the surface, a visibility cone was computed. Ding et al. [40] used the same method for grid points. This method considerably reduces the occlusion problems (m ✓). However, this method only works for laser type sensors. Specifically, the patches of the surface are separated according to only the orientations, which is suitable for laser sensors that scan continuously along a path; however, for certain sensors (e.g., structured light sensors), a well-defined

field must be considered because such sensors scan only a precise area in one acquisition.

Germani et al. [31] and Bircher et al. [41] considered one side of the mesh to represent a segmentation of the surface (q ✓). In this case, the segmentation depended on the mesh size of the object to be scanned. However, this method was efficient only when the subdivision of the mesh was homogeneous, and the number of faces was not extremely large.

Shi et al. [42] divided the mesh into patches according to the face normals of the mesh (q ✓). For each patch, a bounding box was generated to contain all the triangles. This method was similar to those employed by Sadaoui et al. [28], Lartigue et al. [29] and Koutecky et al. [30].

3.2.2 Viewpoint position and orientation optimization methods

In the aforementioned methods, to solve the SCP, the viewpoints were first sampled and later selected from this set of points. With certain exceptions, the approaches based on coverage optimization are different. Instead of choosing from a set of points, the viewpoints are generated individually, and their position and orientation are optimized.

Lee et al. [38] used a laser scanner in their approach. The orientation of the sensor was determined, and the difference in the angle between this orientation and the normal to the surface of each point of the patch was minimized. The distance from the viewpoint to the surface was determined based on the sensor characteristics. For each patch, the scan path was determined by creating a rectangle enclosing the patch and dividing it into subrectangles along the x axis. Likewise, for each patch, Phan et al. [36] determined a scan direction and path by ensuring that the laser ray is perpendicular to the direction of travel of the path. The scan direction was defined as the average of the normals of the faces involved in the scan path on the patch. Similar to Lee et al. [38] and Phan et al. [36], Souzani et al. [39] minimized the sensor orientation such that the corresponding angle with the surface normal was minimized (n ✓). The direction of the path was determined to scan the entire patch, in this case, a voxel, and the distance of the sensor from the surface was determined based on the sensor characteristics. Lee et al. [38] decreased the size of the scanner ray such that each actual scan is larger than the simulated one, and a guaranteed overlap occurred between scans (k ✓). Nevertheless, the approach does not involve any control, and the shape of the surface involved in the overlap remains unclear. Phan et al. [36] generated their viewpoints to control the overlap between each scan (k ✓). The distance between the generated parallel planes on the 2D surface was determined according to the overlap ratio between two desired passes. However, Souzani et al. [39] did not provide any information regarding the consideration of the overlap between scans (k?), and thus, a correct alignment between each measurement could not be ensured.

Germani et al. [43] and Wu et al. [37] defined a viewpoint per patch by minimizing the angle between the normals of the patch and direction of the sensor (n ✓). Ding et al. [40] initialized a direction and searched for all the patches on the surface for which the average normals did not diverge excessively from the initial direction. The directions were generated iteratively until the surface was covered. If the surface could not be completely covered with the generated directions due to surface occlusions, new directions were generated by adjusting the initial ones. Using a similar approach, Lee et al. [32] and Son et al. [33] determined the directions, distances and orientations for each patch. In the presence of occlusions, if the surface of the patch could not be scanned completely (m ✓), the viewpoint was changed, and if this configuration was insufficient, a new viewpoint was added (see Fig. 15).

The disadvantage of this method occurs in the case when the surface is unreachable (e.g. in the event of extremely deep

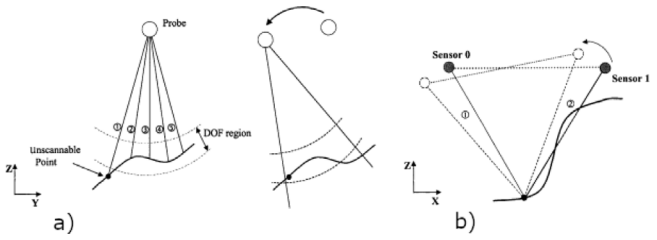

Fig. 15. Modification of a scan direction due to: (a) DOF constraint, (b) occlusion problem [33].

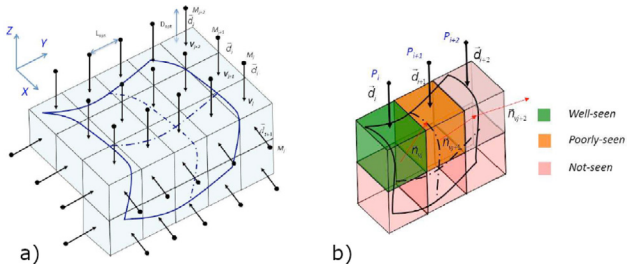

Fig. 16. Representation of the initial set of viewpoints (a), and voxel qualification (b) [29].

holes). In this case, the new viewpoints must be added at infinity, although the maximum number of viewpoint re-orientations and additions must be fixed. Son et al. [33] and Lee et al. [32] used the same approach as that of Lee et al. [38] to ensure an overlap between the scans and decrease the size of the scanner line such that each real scan is larger than the simulated one (k ✓).

Lartigue et al. [29] used the same principle in their Voxel2Scan method. For each patch, a set of six initial directions was defined along the (x, y, z) axes of the grid. The voxels were qualified as "well seen", "poorly seen" and "not seen", depending on the occlusions (m ✓), distances to the sensor, field size, angle of the sensor to the surface (n ✓) or collisions of the sensor with the environment (o ✓), as shown in Fig. 16. Subsequently, for unseen voxels, two strategies were exploited. In the first strategy, new directions were added at the intersection of the planes of the main coordinate system, and the process was restarted with the newly added directions. This process was conducted iteratively until all the voxels were seen. The second strategy was adopted in the case in which the normals varied considerably, and the process resulted in numerous voxels. The positions calculated using the initial voxelization were retained. The parent voxel was considered, and the viewpoint was created from the center of this voxel. However, small occlusions due to the strong segmentation were not processed.

Germani et al. [31], Raffaeli et al. [34,35] and Koutecky et al. [30] used a visibility map to calculate the viewpoints. First, a viewpoint was defined using a point (mean point of the patch) and a vector (mean normal of the points of the patch) for each patch. The position was along the normal at a distance from the mean point, defined by the sensor characteristics. The visibility map was used to determine the occlusion directions. This map was computed for each point of the patch by projecting the faces to be inspected onto a sphere centered on the point (see Fig. 17).

Only the rays having an angle of less than 50° with the surface normal were projected onto the sphere to preserve the high point cloud quality at the time of scanning (n ✓). This visibility map was used to automatically detect an occlusion (m ✓). Subsequently, the sphere was sampled every degree and transposed into a

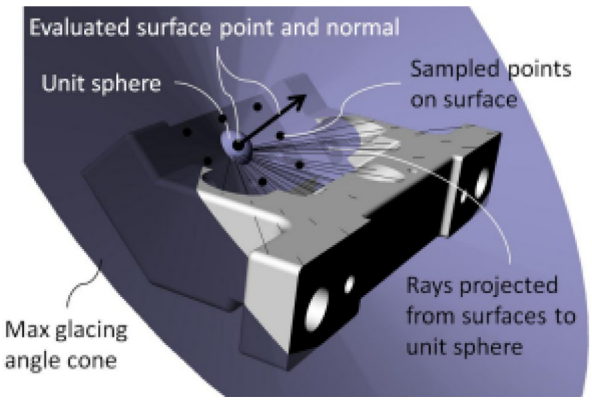

Fig. 17. Determination of the visibility map. For a generic point, the occlusion produced by another surface was evaluated in the cone corresponding to the maximum glancing angle [35].

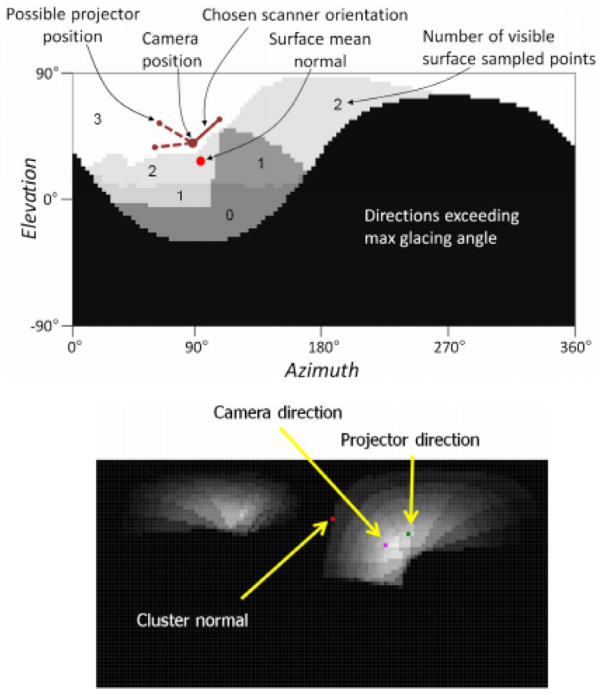

Fig. 18. Example of combined visibility map obtained from the intersection of the fictitious (upper) and real tree points (lower). The red dots represent the mean surface normal, and the different tones of gray indicate the regions with different levels of occlusion. The scanner orientation is searched in the most white area [35]. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

matrix. The viewing maps were assembled (overlapped) to produce a grayscale image (see Fig. 18). The scanner positions were searched in the areas with the maximum number of visible points to maximize the overlap of the patches. The disadvantage of this type of map is the large time required for the determination. The calculation cannot be performed in real time and must be performed before the execution of the algorithm.

The approach of Bircher et al. [41] did not focus on minimizing the number of viewpoints. Similar to the methods for PCS resolution, in this approach, the viewpoints for each face of the object's mesh were generated. The principle was to solve the problem of the commercial traveler, that is, to minimize the travel time between all the viewpoints (h ✓). The viewpoints were sampled to minimize the angular distance between the normal of the face and the orientation of the sensor (n ✓), and to minimize the distance to the neighboring viewpoints. To this end, the *TSP solver* Lin–Kernigen–Helsgaun (LKH) heuristic [44] was used to calculate the optimal path. This method could minimize the time taken for the reconstruction by calculating a "minimum path" between each viewpoint. However, no notion of the overlap between the viewpoints was considered. Consequently, the model was required to be convex, or the cavities were required to be sufficiently large for the surface to be scanned entirely.

The principle of *feedback* was employed by Shi et al. [42]. This method is somewhat similar to the so called *search based* methods, which are not aware of the model and whose objective is to find the next viewpoint sequentially, based on the part that has already been scanned. Shi et al. [42] start with an initial mesh of the preconstructed object, parts of which were missing in the complete covering. The authors initially generate the viewpoints considering the enclosed boxes of the patches and update them based on the result of a cost function that check whether the holes in the mesh were filled. Here again, holes correspond to part of the object that have not been acquired. If a stop criterion is not defined, the process could add infinite dots to fill the holes. The criterion corresponds to the ratio between the total area of the holes, and the measurement error between the model and resulting point cloud. To ensure that the process ends satisfactorily, a maximum number of iterations can be defined (e ✓). Numerous stop criteria could be used depending on the objective being maximized. The measurement quality and overlap are prioritized over the number of points (g ✓, n ✓). If only the overlap is being maximized, a ratio between the total area of the surface and the total size of the remaining holes can be considered. Moreover, it could be verified if the newly generated viewpoints add new information (e.g. area of holes not seen previously) to avoid unnecessary iterations.

The method of Sadaoui et al. [28] was slightly more unique in that it combined laser scanning and probing. The calculated viewpoints were a set of orientations for the scanner and the probe. The set of orientations and positions for the scanner was defined according to the method of Souzani et al. [39] (m ✓, n ✓). The orientations for the probe were calculated using the method proposed by Cho et al. [45]. These orientations corresponded to particular orientations defined using the touch probe approach orientation (PAO) method for features such as cylinders, spheres, planes, and cones. The viewpoints were selected using an accessibility function for the scanner and probe. The advantage of combining the two measurement techniques was that the entire surface to be measured was ensured to be reachable. However, the inspection time reduced through optical measurement was lost owing to the time spent performing measurements using the probe. In addition, the data processing using the two techniques could not be identical, and thus, the data was required to be processed separately. Specifically, the touch probe considers only a few extremely precise points on the surface to be inspected, while the scanner measures thousands of points with a lower accuracy. Moreover, the two devices do not work in the same reference frame, and a registration with the ICP algorithm cannot be realized to reconstruct the entire model later.

Most of these methods used generalizable and configurable viewpoints for any type of sensor, as indicated by Souzani et al. [39], Shi et al. [42], Germani et al. [31,43], Raffaeli et al. [35], Lartigue et al. [29], Bircher et al. [41], Wu et al. [37] and Koutecky et al. [30], who defined, for each viewpoint, a position and an orientation in a global reference frame (a ✓). Certain methods took into account the particularity of the laser scanner, and instead of defining a position for the sensor, a scan direction associated with a distance from the surface was defined. Although this method is not entirely generic, it can be easily adapted to a higher generalization by dividing the scan path into several positions along the

path direction, depending on the orientation of the sensor and the distance to the surface, as indicated by Lee et al. [32,38] and Son et al. [33] (a ∼). Ding et al. [40] also used viewpoints with a sensor specific orientation and scan direction (a ∼). However, the information regarding how the distance to the surface is calculated, or whether the surface is within the sensor's field of view is not unclear. If these aspects cannot be verified, the method can be applied only for small objects, for which all the surfaces with the same direction are within the scanner's field of view. The method of Phan et al. [36] does not define a real viewpoint, but a trajectory composed of a set of couples of piloted points/orientations for a given direction. The direction of the trajectory is defined by the length of the rectangle of the 2D surface, and the point pairs are sampled on each edge of the mesh along the defined scan path. This type of definition is therefore not generalizable to all types of sensors. Another method that cannot be generalized in this manner is that of Sadaoui et al. [28], which takes into account two types of viewpoints, one of which is specific to the CMM probe.

One of the advantages of the approaches based on coverage optimization is that, since the viewpoints are calculated and optimized directly according to the surface, the methods are a priori adaptable to all object sizes (b ✓), except the approach of Ding et al. [40], for which the information is not available (b?).

3.2.3 Scan number computation methods

The methods described in this section do not seek to minimize the number of scans at all costs, but rather, to maximize the surface coverage. Consequently, many methods consider that defining a scan by a patch is a sufficient solution. Lee et al. [38], Germani et al. [31,43], Phan et al. [36], Souzani et al. [39], Bircher et al. [41] and Wu et al. [37] adopted this type of solution (g ✓).

Certain other methods initially define a scan for each patch and later adapt the viewpoints, or the segmentation of the surface, according to the parts observed or not observed. Lee et al. [32], Son et al. [33] and Ding et al. [40] computed a viewpoint for each patch and evaluate all the viewpoints. If the entire surface of the patch was not visible, the position and orientation of the viewpoint were changed. If this step was insufficient, a new viewpoint was added (g ✓). Shi et al. [42] re-evaluated the viewpoints of each patch at each iteration with a cost function and added or modified the viewpoints according to the result obtained. The use of such a function allowed the easy addition or removal criteria according to the needs (d ✓), similar to the methods involving the transposition to the SCP or search based methods. Lartigue et al. [29] and Koutecky et al. [30] also defined a viewpoint per patch and directly modified the segmentation of the model according to the result to ensure maximum coverage of the surface (g ✓).

Raffaeli et al. [34,35] segmented the surface into patches and computed the combined visibility maps of each patch; consequently, one viewpoint per patch was calculated. The viewpoints were evaluated and ranked in ascending order of the number of points covered on the surface. Subsequently, the views at the bottom of the list and those that did not add new points to be covered were deleted. Later, a new iteration of the process was launched with all the points that had not been covered. This process is repeated until it was no longer possible to add new viewpoints that covered the unseen points on the surface (g ✓). Using this method, the list of viewpoints was continuously optimized by adding better viewpoints and removing the less effective viewpoints. However, the process was stopped after a number of iterations, because certain points could never be accessed, rendering the algorithm to loop endlessly. Raffaeli et al. [34,35] did not use the information from visibility maps to predetermine the completely occluded points, and, therefore, removed them from the viewpoint coverage assessment computation.

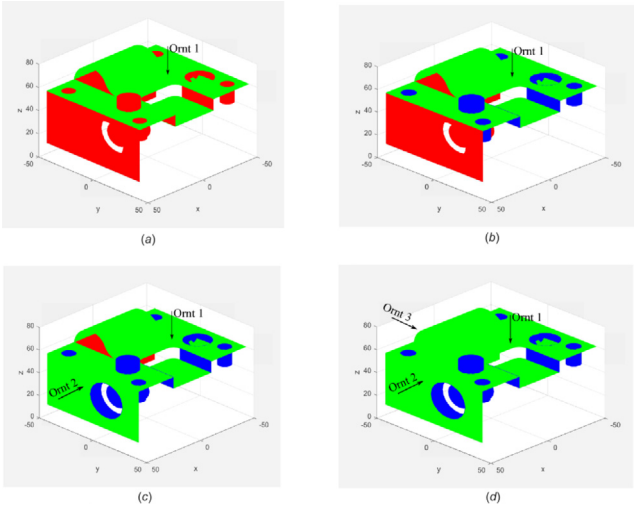

Fig. 19. Result of an operation sequence generation: (a) laser ability evaluation at the first iteration, (b) after one iteration, (c) after two iterations, and (d) after three iterations. The green and blue regions indicate the surfaces measured using the laser sensor and touch probe, respectively, and the red region indicates the non-measured surface [28]. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Sadaoui et al. [28] established a list of viewpoints with both sensor and probe orientations. The method ordered the list according to the accessibility of the sensor and probe and iteratively tested the viewpoints in the list. At each iteration, the viewpoint was evaluated, i.e., the surfaces scanned by the laser scanner were examined, and subsequently, the area around the surfaces that could not be reached by the scanner but by the probe was examined. At the end of the iteration, a group of features corresponding to this viewpoint was created. The process was repeated for the following viewpoints until all the features were reached from one viewpoint, as shown in Fig. 19. (g ✓).

All these methods involve an iterative process that can end autonomously (e ✓). In certain cases, if the algorithm ends when the surface is not completely covered, new viewpoints are manually added, as in the approaches used by Germani et al. [43] and Raffaeli et al. [34,35]. This aspect makes the process semi-automatic instead of fully automatic (e ∼).

After calculating the optimal number of scans, certain methods try to minimize the total measurement time, by minimizing the travel time between each scan. This problem is transposable to the traveling salesman problem (TSP), another optimization problem that involves finding the shortest Hamiltonian cycle in a graph. In the context of the SCP, it has been demonstrated that the problem of nonoriented Hamiltonian cycles is an NP complete problem. In such approaches, the resolution of this problem is not paramount, and thus, heuristics are used. Germani et al. [31,43] and Raffaeli et al. [35] used Dijkstra's algorithm to search for the shortest path in a graph based on pairs of scans that have an overlap (h ✓) of a high quality. Such methods attempt to identify the absence of a small curvature in the overlap (l ∼). Koutecky et al. [30] used a modified solution of Shintyakov's TSP [46], which takes into account the distance between the positions, angle between the left camera and surface normals and angle between the x axis and surface normals (h ✓). Bircher et al. [41] attempted to solve the TSP problem by using the LKH heuristic [44], as discussed in Section 3.2.2 (h ✓).

3.2.4 Algorithm validation

Most of the approaches discussed in this section correspond to applications of inspecting the mechanical parts. Compared to

Fig. 20. Firefly UAV equipped with the VI sensor [41].

other methods, the coverage optimization approaches validate the algorithms on industrial parts with rather complex geometries. Specifically, Koutecky et al. [30], Sadaoui et al. [28], Germani et al. [31,43] and Ding et al. [40] validated their methods on such parts (p ✓). Raffaeli et al. [34,35] validated their method considering the inside of a car door, and Shi et al. [42] performed the validation considering a part from the inside of a car with certain small variations (p ✓). Lartigue et al. [29] worked on a crankshaft using the approach of Zuquete Guarato [47] (p ✓). This validation demonstrated that the developed algorithms can be industrialized. In each case, the accuracy of the resulting point cloud was considered, and thus, measurements on these clouds could be realized.

Several of the presented approaches correspond to preliminary work aimed at an application on a mechanical part, and the validation is conducted on parts with simple geometries, for example, on a half cylinder or planar shape with small variations in the curvature. Lee et al. [32,38], Son et al. [33], Phan et al. [36] and Wu et al. [37] validated their work on such academic parts. These validations cannot be used to determine whether a specific method can be transposed to real industrial parts with more complex shapes.

Souzani et al. [39] validated their work on objects not intended to be measured afterwards (e.g. small cars and figurines) (p NA). Bircher et al. [41] developed a method to be applied to large objects without any high precision requirements for the reconstruction of the complete point cloud. Their method was validated on monuments (i,p NA), and it was not clear whether the approach could be applied to small industrial parts. The authors developed the tool used to validate the method, which included an UAV equipped with a sensor with two cameras (see Fig. 20).

The main sensor types used in these methods include the laser scanner and structured light sensor. In the case of a laser scanner, a single line is projected, and it is necessary to move the scanner to scan the part; therefore, the considered viewpoints have a scan direction (or path) associated with an orientation. In contrast, for the structured light sensor, a position is associated with an orientation, because in this case, a single measurement allows a point cloud to be retrieved from an entire model. The size of this measurement is defined by the characteristics of the field associated with the sensor. Lee et al. [32,38], Son et al. [33], Souzani et al. [39], Ding et al. [40] and Phan et al. used such approaches involving a laser scanner. [36]. Sadaoui et al. [28] also used a laser scanner, although a probe was used in combination. Lartigue et al. [29] proposed a method that could be applied to both laser scanners and structured light sensors and validate this method considering the two measurement means. Germani et al. [31,43], Raffaeli et al. [34,35], Shi et al. [42], Wu et al. [37] and Koutecky et al. [30] used the methods involving structured light sensors, as shown in Fig. 21.

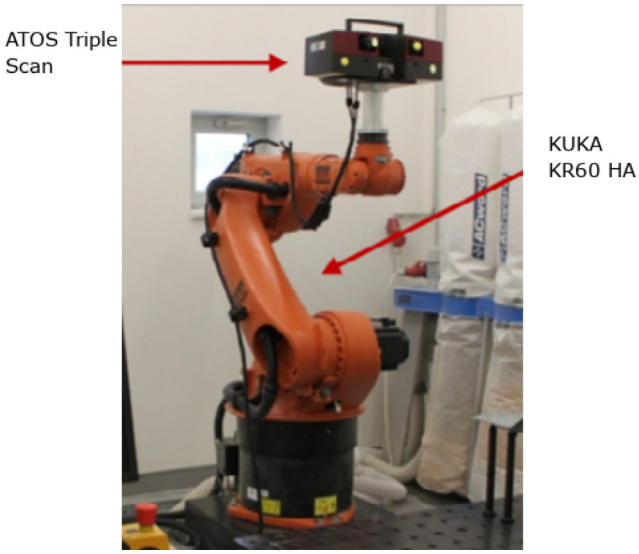

Fig. 21. Configuration involving a KUKA robot arm and an ATOS structured light sensor [30].

The association of a robot arm with a structured light sensor is widely applied, because only one robot position per scan is required. In this context, the low precision positioning of the robot does not lead to errors in the point cloud resulting from a measurement. Germani et al. [31,43], Raffaeli et al. [35], Wu et al. [37] and Koutecky et al. [30] used such a configuration to validate their work. Laser scanners, which measure only one line at a time, require a support to ensure the precise positioning of the scanner to have a coherent point cloud. To this end, Phan et al. [36] combined an optical tracker with the laser scanner/robot arm couple, thereby allowing a more precise positioning of the scanner compared to that achieved by the robotic arm. Another widely used configuration is the laser scanner/CMM combination. The CMM is a tool used in metrology, associated with the use of probes to measure parts precisely. A laser scanner mounted on a CMM is therefore an ideal configuration to reconstruct a point cloud highly accurately. Son et al. [33], Souzani et al. [39], and Lartigue et al. [29] employed this configuration, in which both the laser scanner and structured light sensor were combined with the CMM. In addition to associating the laser scanner with the CMM, Sadaoui et al. [28] maintained the classic association of the CMM with a probe (see Fig. 22).

3.2.5 Summary of coverage optimization methods

The so called coverage optimization approaches were described in this section. Unlike SCP methods, these approaches do not seek to minimize the number of viewpoints at all costs, but rather, to optimize the covered area. A generic scheme of the four main steps forming the core of these methods was formalized.

In the first step, as in the case of the SCP transposition approaches, a segmentation must be performed, and three methods can be used to this end. Lartigue et al. [29], Sadaoui et al. [28], Koutecky et al. [30] and Shi et al. [42] adaptively voxelated their model using different methods. Lee et al. [32,38], Son et al. [33], Raffaeli et al. [35], Souzani et al. [39] and Ding et al. [40] sampled the surface and cluster it into subsurfaces by using mainly the mesh normals. Germani et al. [31] and Bircher et al. [41] considered one side of the mesh to represent a patch on the surface.

The next step in this type of method was the construction of the viewpoints. In such methods, the viewpoints were not

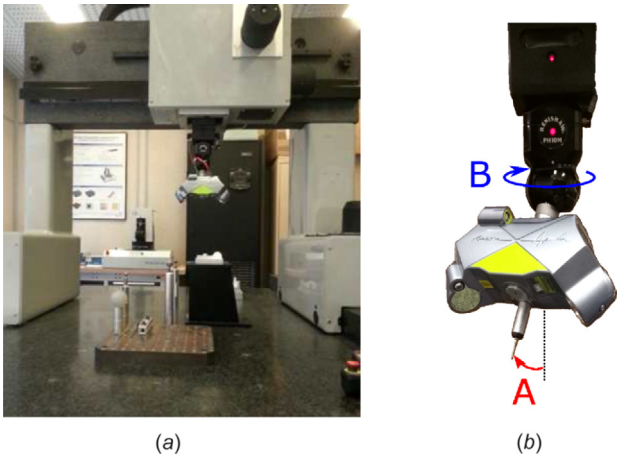

Fig. 22. Configuration involving a CMM and hybrid sensor: (a) view of the CMM, (b) scanner KA50 composed of a touch probe and laser sensor mounted on a Renishaw PH10 head.

sampled and later selected. In fact, the viewpoints were computed directly from the surface segmentation. Consequently, the previous segmentation step was a key step in this method. The way to define a viewpoint was highly specific to each method. In contrast, the computation of the number of viewpoints was common to most methods. In the approaches of Lee et al. [38], Germani et al. [31,43], Phan et al. [36], Souzani et al. [39], Bircher et al. [41] and Wu et al. [37], a viewpoint was optimized for each patch. Lee et al. [32], Son et al. [33], Ding et al. [40], Lartigue et al. [29], Koutecky et al. [30] and Raffaeli et al. [34,35] optimized a viewpoint per patch and later adapted the segmentation if necessary, or added a new viewpoint. Only Shi et al. [42] proposed a method that approximated the viewpoint based on solving the SCP.

These methods, unlike the SCP methods, are not optimized to easily add or remove constraints. In SCP methods, the constraints are mainly included in the cost function. In contrast, in these approaches, the implementation of the constraints was split between the segmentation and optimization of the viewpoint position.

4 Search based approaches

Search based methods do not have knowledge regarding the model to be scanned (i NA, j NA). These algorithms are generally iterative, and within each new iteration, the algorithms seek the next optimal scan. In each iteration, a space evaluation is performed to determine an area in which the surface could "probably" be the most suitable. Four main steps are common to all these methods, which resemble those of the approaches involving the SCP transposing. A cost function is maximized at each iteration, albeit the stopping criterion is not the coverage of the model, because it the model is unknown. First, a representation of the surface $S^{[0]}$ seen from an initial viewpoint $pv^{[0]}$ and void volume $P^{[0]}$ is formulated. Subsequently, a sampling method of the viewpoints is applied. A $Ps^{[k]}$ list of viewpoints is sampled at each iteration k. A viewpoint pv is generally described with a vector pos which represents a (x, y, z) position and an orientation (rx, ry, rz), and a vector w_{pv} which contains the patches p_i of the object surface belonging to P visible from pv. At the beginning of the method, the vector w_{pv} is empty. The objective function to be optimized is established in order to maximize the set of new patches p_i seen by the view point pv and its w_{pv} vector at the current iteration k. A stopping criterion related to the number of newly added patches p_i is used to stop the iterative process (see Fig. 23).

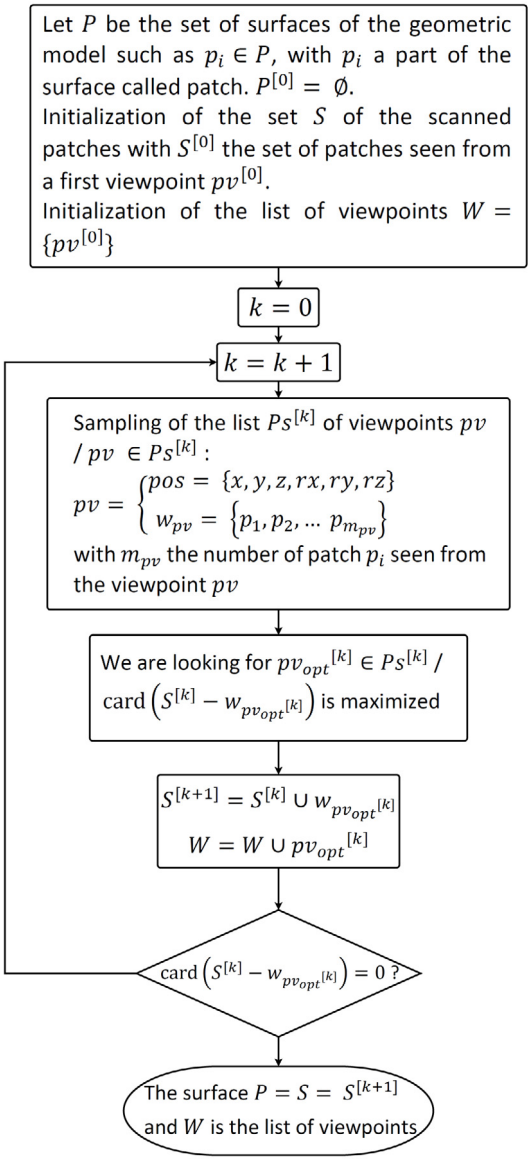

Fig. 23. Formalization of the resolution steps for search based approaches.

4.1 Observed surface and void volume representation

One of the difficulties in such algorithms pertains to obtaining an efficient representation of the volume of both the scanned and nonscanned parts. Pito [48,49] represented the observed surface by using a simplified mesh of the scanned parts, in which the edges of the mesh were the original edges of the unsimplified mesh (q ✓). The nonscanned space was represented in a more complex manner. Only the empty volume near the edges of the scanned area was represented as a small rectangular patch. By considering the orientation of the sensor when scanning, the space can be divided into three types: the observed space, the empty space around each edge, and the unknown space (see Fig. 24). The advantage is that the void patches are in the continuity of the surface, and thus, when scanning a void patch, the algorithm is constrained to have an overlap with the surface already scanned (k ✓).

To represent the observed surface, Loriot [8] relied on the mass vector chain (MVC) developed by Yuan [50]. The model was considered as a convex object. The method was based on the fact

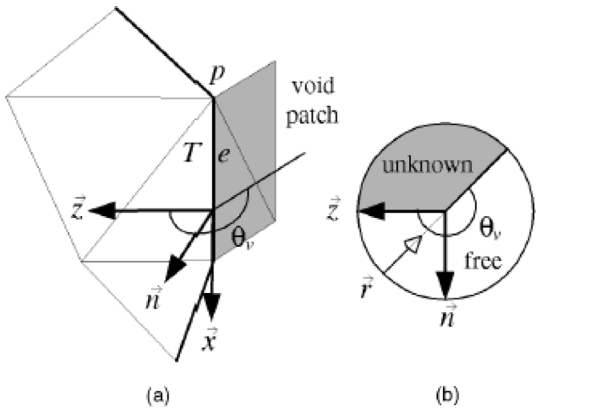

Fig. 24. Efficient representation of the scanned and nonscanned areas: (a) coordinate system of an edge, (b) free space map induced by the scan orientation r [49].

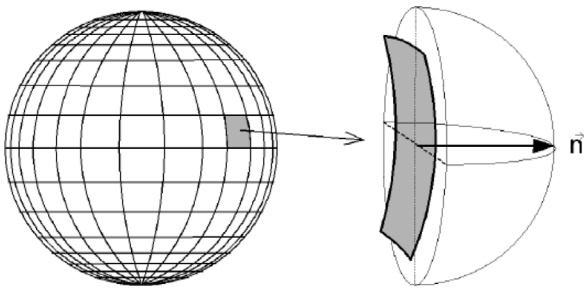

Fig. 25. Intermediate positioning space divided into two subspaces: PSS and PSD [49].

that, for a convex object, the total Gaussian mass must be equal to zero. The sum of the normals of the scanned surface provided a vector corresponding to the SVM of the current view, and the inverse of the vector provided the direction of the nonacquired surfaces. This step considered the object only as a convex shape that cannot exist autonomously. Therefore, at the end of this step, the type of result obtained was incomplete if the part was not convex. Consequently, in the second step of his method, Loriot identified the "holes" in the mesh as missing data to be acquired, and thereby, to be covered (q ✓). Kriegel et al. [51] also used this notion of holes (q ✓).

To represent space, voxel grids are commonly used, as in the approaches of Kriegel et al. [51,52] and Vasquez-Gomez et al. (r ✓); even probabilistic voxelated space has been employed in certain cases [53,54]. This voxel grid represents the probability of a cell being occupied. At the beginning of the process, the grid is empty, and it is filled each step. In 2009, Vasquez et al. [55] used a voxel grid (r ✓); however, instead of labeling voxels with a probability of occupancy, the voxels were directly labeled using five states: unmarked, occupied, empty, occluded, occplane (i.e. adjacent with none of the six faces of an empty voxel). The authors narrowed the labels to three states in their subsequent work in 2014 [56]: occupied, free and unknown.

By using a grid of voxels to represent the space, other objects in the scene can be detected, thereby potentially avoiding collisions with the sensor (o ✓). A second advantage of this type of method is that a nonbinary type can be assigned to a part of the space. The areas of space represented by the voxels are not only seen/unseen, the unseen parts are differentiated from those that could be on the surface and are obscured by the object itself.

4.2 Viewpoint sampling methods

The viewpoint sampling methods are applied at each iteration of the algorithm. Subsequently, the viewpoints are evaluated, and the viewpoint that maximizes the cost function is selected as the next optimal scan. The sampling method is therefore a key step of the algorithm.

Pito et al. [48,49] used an intermediate positioning space divided into two subspaces. The first space corresponded to the positional space surface (PSS), which encompassed the volume of the object to be viewed. The second space represented the positional space directions (PSD), which encoded the directions of the observation beams for each point of the PSS (see Fig. 25).

The viewpoints were transformed into the P(w, y, α, θ) space, in which the PSS was represented in terms of w and y, and the PSD was represented in terms of α and θ. An image of each viewpoint x_i of X was computed in the PS. The viewpoints were sampled around a circle at an angle that was a multiple of 4; the barycenter was the center of the object, and the radius was defined according to the parameters of the sensor used. This type of sampling is not optimal and does not allow many degrees of freedom; specifically, only a rotation around a single axis was available in this case. Consequently, concave objects larger than the focal length of the scanner cannot be scanned, and, depending on how the scanner is mounted, the top or bottom of the part may not be scanned. Moreover, this way of defining the viewpoints is not generalizable to all types of sensors.

Loriot et al. [8] thus employed two steps. The first step did not require sampling the viewpoints, because the viewpoints were directly determined according to the VMCs at each iteration (see Section 4.1). The second step involved filling the remaining holes. For each identified hole, a normal was calculated, and the viewpoint was fixed and oriented around this normal. If an occlusion appeared, a sphere was sampled and centered on the hole. Each point of the sphere included in a 60° cone around the normal of the hole represented a potential viewpoint to scan the considered hole. As in the previous method, although the sphere allowed higher positioning freedom than that allowed by a circle, if the hole to be filled in had a radius greater than the distance recommended by the scanner, the surface was likely never scanned because the scanner in this case would be extremely far from the surface. However, one of the advantages of this method is that several types of solutions can be attained before trying to sample the viewpoints to optimize the next optimal view when filling holes. Moreover, the holes to be plugged in the case of the object presented in the method may not be sufficiently large to present this type of configuration (b ∼).

Vasquez et al. [55,56] also used a sphere to sample viewpoints, with the sphere centered at the barycenter of the object to be scanned. Although this solution can be easily implemented, it involves the same limitations as those presented above. The solution of sampling the viewpoints around a sphere only works in the case of objects smaller than the measurement range of the sensor used.

Kriegel et al. [51,52] sampled viewpoints along the edges and holes of the mesh at each new search. The advantage of this method is that, for each potential viewpoint, there exists a minimum overlap with the already scanned areas (k ✓). The viewpoints computed by Kriegel et al. are represented as scan paths with a start and end point, an orientation, a direction and a distance to the model (see Fig. 26). Subsequently, the viewpoints are evaluated such that the scan paths are collision free (o ✓), and if the surface is occluded by an environmental obstacle, the viewpoints around the object are rotated until the occlusion disappears (m ✓). The advantage of using this method to sample viewpoints along the edges and holes of the mesh is that it allows the algorithm to be applicable for objects of all sizes (b ✓).

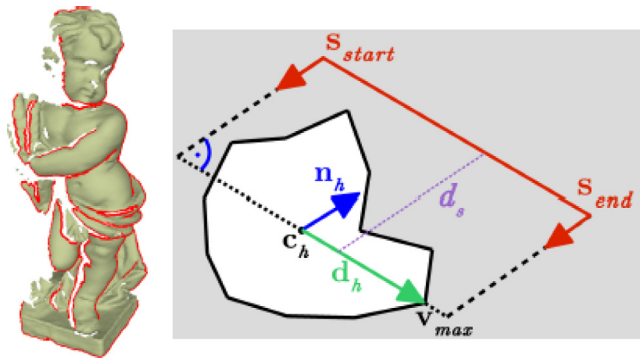

Fig. 26. Viewpoint sampling method: (left) boundaries detected in partial meshes of the Putto statue; (right) a scan path in the direction of the largest expansion of the hole d_h, in the inverse hole normal n_h direction and at the optimal sensor distance d_s [51].

Vasquez et al. [53,54] sampled viewpoints directly in the robot's configuration (kinematics) space (c ✓). The viewpoints were sampled randomly with a uniform distribution at each new iteration. Subsequently, the viewpoints were evaluated to ensure that they do not collide with the environment (o ✓) and that the orientation radius of the sensor intersects the box surrounding the object. Next, the visibility of the viewpoint was determined (m ✓) to evaluate the percentage of overlap (k ✓). Viewpoints that did not meet the minimum criteria were removed from the set of candidate viewpoints. The advantage of this method is that the viewpoints can be generated directly according to the sensor support, in this case, a robotic arm. This principle allows the viewpoints to be determined automatically with collision free paths between each scan, thereby helping reduce the inspection times. However, this method only works if the support can provide this type of information automatically. Moreover, the approach is not generalizable to all configurations and types of sensors, and the size of the object to be reconstructed is constrained by the maximum extension of the robot arm (b ∼).

In contrast, the methods of Loriot [8], Vasquez et al. [55,56] and Kriegel et al. [51,52] define the viewpoints in a general way and can thus be generalized to all types of sensors and supports (a ✓).

4.3 Stop criteria

The process of searching the next optimal view is iterative. Therefore, a stop criterion must be established to maximize the quality of the reconstruction while ensuring that the algorithm terminates autonomously in a reasonable time (i.e. in a time competitive with that of human computation). The stop criteria are similar in most works and involve achieving a certain percentage of surface coverage (g ✓). Pito et al. [48,49] and Loriot [8] used a so called redundancy criterion. At each iteration, the rate of the new surface covered was calculated and, if a certain percentage of the new surface covered was not reached, it was considered that the algorithm could not find any new surface. Subsequently, the algorithm terminated. (e ✓). Similarly, Vasquez et al. [53–56] used a surface factor that provided information regarding the rate of unknown voxels observed by the viewer. If the factor was below a certain threshold, the process was terminated. Vasquez et al. considered that adopted a criterion based solely on the model was insufficient. A second criterion was applied, and the process was terminated if the robot did not identify any path between its current position and the positions of the candidate viewpoints (e ✓).

Kriegel et al. [51,52] computed a coverage index based on the mesh and holes detected in the mesh, as well as, the average density of the points that the algorithm was required to reach before it terminated. If none of the criteria were satisfied, as in most methods, the process terminated when a predefined maximum number of scans was attained (e ✓).

4.4 Objective function to be minimized

The objective function (or cost function) is a key parameter of this type of method. The principle of these methods is that, at each iteration of the process, a viewpoint is chosen as the best possible next scan among a set of possibilities. To this end, the viewpoints are evaluated according to a cost function, and the viewpoint that maximizes this function is chosen as the next optimal scan. Each method involves its own objective function that meets well defined criteria.

Pito et al. [48,49] established an objective function that sought a viewpoint that maximized the coverage of the empty patches, while maintaining a certain ratio of the area already scanned. To this end, the rate of the new visible area was prioritized while maintaining a ratio of the area already observed, such that an overlap existed between each scan (k ✓). The overlap between the next optimal scan and the area already observed was also a key criterion for the methods used by Kriegel et al. [51,52] and Vasquez et al. [53–56] (k ✓).

Kriegel et al. [51,52] determined an objective function with two components. The first component represented the "exploration" part, i.e. considering the viewpoints that maximized the number of voxels with the greatest information gain (IG). The algorithms attempted to search for the viewpoint that maximized the sum of the probabilities of the voxels visible from it. The probability of a voxel represented the probability that the voxel was occupied, thereby indicating that this part of the object was required to be scanned. The second component represented the "modeling" part, in which the viewpoint that could observe the voxels with a high quality was selected. To compute the acquisition quality of a voxel, the angle between the normal to the surface and sensor orientation was calculated. The advantage of this method was that the unknown regions, overlapping a surface already scanned, could be scanned, and an area considered to have a low quality could be rescanned (k ✓, n ✓).

This notion of the quality of acquisition was also employed by Vasquez et al. [55,56], whose objective function was formulated considering four criteria. As mentioned previously, the quality of acquisition was considered in terms of the angle between the normal to the surface and sensor orientation (n ✓). The main criterion was a factor that considered the rate of the new surfaces with the rate of the already observed surfaces, such that the selected viewpoint overlapped with the already observed surface. The third criterion corresponded to a navigation criterion, which helped minimize the geodesic distance between the next optimal view and the previous scan to minimize the travel time, and thus, the inspection time (h ✓). The last criterion was an occlusion factor that examined whether the surface was occluded or not from the considered viewpoint (m ✓).

In 2014 and 2017, Vasquez et al. [53,54] used a completely different objective function, for which the method viewpoints were sampled directly in the robot workspace. This function also considered four criteria. The first criterion ensured that the viewpoint did not collide with the environment, and that the path between the previous scan and the viewpoint was collision free (o ✓). The second criterion corresponded to the degree of overlap between the area covered by the viewpoint and the area already scanned (k ✓). The third criterion corresponded to the ratio of the new voxels not yet observed, but visible from the current

viewpoint, to the total number of voxels not observed. The last criterion corresponded to the distance between the previous scan and the viewpoint, with this distance computed according to the robot's degrees of freedom. The weights associated with the axes were determined experimentally such that the robot motion time was minimized (h ✓). The latter principle can be both an advantage and disadvantage of this method. When a robot arm support is used, this aspect can prevent the robot from conducting excessively large movements if a position with a shorter axis movement is possible. This condition can minimize the robot movements, thereby reducing the inspection time. However, this method only works when using a robot arm, and therefore, the calculation of the viewpoints cannot be generalized to any type of support.

An advantage of the methods of Vasquez et al. and Kriegel et al. is that the selected objective functions allow many criteria to be considered in a modular way. The addition, modification, or removal of new criteria is convenient in such approaches (d ✓).

Loriot [8] established an iterative process involving several steps. The first step was to use the VMCs to perform an initial reconstruction of the part. In the second step, the process attempts to cover all the holes of the mesh, considering the areas not scanned in the first step. During this step, the viewpoint is first positioned along the normal of the hole and oriented toward this direction. Subsequently, the viewpoint is evaluated, and if self occlusion occurs, the viewpoint is sampled. The sampled viewpoint that maximizes an objective function is selected (m ✓). The objective function employs a single criterion to ensure the selection of the maximum number of visible points of the hole. One of the disadvantages of this method is that in the first stage, a common surface may not exist between two scans. Consequently, the alignment between scans is not ensured at this stage.

None of the algorithms of this category can evaluate the quality of the overlap between scans. Specifically, these algorithms can assess the sufficiency of a surface in ensuring the alignment between two scans, if the two scans share a common surface. However, if the shared surface is extremely small, an algorithm such as the ICP cannot be used to ensure a correct alignment, even if the shared surface has no particular shape and is smooth. This aspect is a key limitation of such approaches.

4.5 Algorithms validation

Several of the methods presented in this section are not based on an input model. Therefore, no a priori knowledge regarding the model to be scanned is available. In the industry, one of the most important applications is reverse engineering, which consists of scanning a part and later reconstructing the CAD model from the point cloud.

Pito [48,49] performed tests on small simple objects such as a telephone receiver, coffee cup and small statue. The type of scanner used to validate this method was a laser scanner mounted on a fixed support. The part to be scanned was placed on a turntable, and it could thus be rotated. Pito's work can be seen as preliminary work, as this type of test does not allow industrialization of the method, because the constraints of the algorithm linked to the sensor are extremely stringent, and the tested objects are extremely simple.

Kriegel et al. [51,52] also used a laser scanner mounted on a robotic arm. In this work, the reconstructed objects were small and midsized, for instance, a statuette, a cake box, and an industrial valve type part. The diversity of the models tested suggests that the method can be used for all types of objects. However, as the result of the method is not intended to be used to perform subsequent measurements, the accuracy of the reconstruction of the complete point cloud is not emphasized.

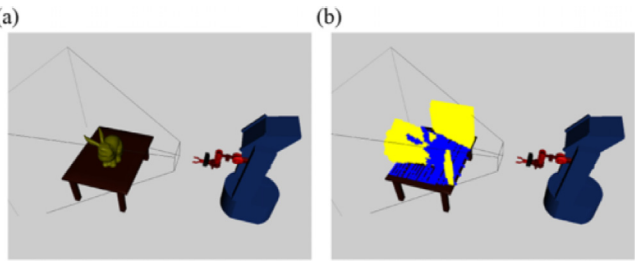

Fig. 27. Stages of the Stanford Bunny reconstruction. The unknown and known voxels are indicated in yellow and blue, respectively [57]. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

In their early works, Vasquez et al. [55,56] did not validate their results on a physical test platform; specifically, only computer simulations were performed for the validation. Consequently, the scanner was simulated and not attached to any kind of support. The simulations were performed on small simple objects such as the Stanford Bunny, a sphere or a coffee cup. This configuration did not allow a full appreciation of the environmental constraints such as collisions or constraints induced by the sensor used.

Nevertheless, in 2014 and 2017, Vasquez et al. [53,54] used a Kinect device attached to a robotic arm with eight axes of freedom. As in their previous work, the algorithm was tested in a virtual environment on small simple objects (see Fig. 27), although the algorithm was later tested on a real platform. The method was validated on objects sized similar to an office chair. As in the case of the existing studies, the objects on which the algorithms were validated were simple, and the method could thus not be directly industrialized.

Loriot [8] used a structured light sensor fixed on a robotic arm to validate his work. The type of object used to validate the method did not differ considerably from the types of objects used in other methods. For instance, the selected objects likely included a coffee cup, small statuettes or industrial parts such as car rims.

Finally, in all methods, the tests performed did not provide information regarding the accuracy of the reconstruction. Without knowledge of the real object, it is not possible to quantitatively validate the accuracy of the reconstruction.

4.6 Synthesis of search-based approaches

The so called search based approaches traverse the surface to be digitized without a priori knowledge regarding the model of the object. As in the case of the previous methods, a global operating scheme common to the algorithms was formalized.

First, an iteratively evolving representation of the space is established. To this end, two groups of methods can be used. Pito [48,49] divided the space into three parts: an observed part represented by the surface, a free part represented by the edges of the mesh and an unknown part. The same concept was employed by Kriegel et al. [51,52] and Vasquez et al. [53–56], although in their approaches, the time space was voxelized and later labeled.

In the second step, the viewpoints were sampled following one the three methods. Pito [48,49], Vasquez et al. [55,56] and Loriot [8] sampled their viewpoints on a surface such as a sphere or circle. Kriegel et al. [51,52] sampled the viewpoints along the borders of the mesh. Vasquez et al. [53,54] sampled the viewpoints directly in the robot's kinematic space.

Contrary to the approaches based on a transposition of the SCP, the process of search based methods is iterative because at

Table 2
Positioning of the approaches with respect to the technological criteria.

Types of method	References	Sensor type			Robot type					Application type			
		Laser scanner	Structured light sensor	Other	Turn table	Robot arm	CMM	Drone	AGV	Small unknown	Large unknown	Small known	Large known
SCP transposition	Scott et al. 2001 [10]		✓				?			✓			
	Scott 2002 [9]		✓				?			✓		✓	
	Martins et al. 2005 [19]	✓					✓					✓	✓
	Scott 2009 [12]	✓					?					✓	
	Loriot 2009 [8]		✓				?			✓		✓	
	Krause et al. 2011 [21]			✓			?					✓	x
	Mahmud et al. 2011 [11]	x					✓					✓	
	Jing et al. 2016 [13]			✓				✓					✓
	Hepp et al. 2017 [20]			✓				✓					✓
	Jing 2017 [14]		✓	✓	✓	✓		✓				✓	✓
	Jing et al. 2018 [15]			✓				✓					✓
	Jing et al. 2018 [16]		✓			✓						✓	
	Mohammadikaji et al. 2018 [17]	✓					?					✓	
Covering optimization	Lee et al. 2000 [38]	✓			✓							✓	
	Lee et al. 2001 [32]	✓			✓							✓	
	Son et al. 2002 [33]	✓			✓		✓					✓	
	Souzani et al. 2006 [39]	✓					✓					✓	
	Shi et al. 2007 [42]		✓		✓					✓		✓	
	Germani et al. 2009 [31]		✓			✓						✓	
	Germani et al. 2010 [43]		✓		✓	✓						✓	
	Raffaeli et al. 2013 [35]		✓			✓						✓	
	Lartigue et al. 2014 [29]	✓	✓				✓					✓	
	Bircher et al. 2015 [41]			✓				✓			✓		
	Wu et al. 2015 [37]		✓			✓						✓	✓
	Koutecky et al. 2016 [30]		✓			✓						✓	
	Ding et al. 2016 [40]	✓					?					✓	
	Phan et al. 2018 [36]	✓				✓						✓	
	Sadaoui et al. 2018 [28]	✓		✓			✓					✓	
Search-Based	Pito 1997–1999 [48,49]	✓			✓					✓			
	Loriot 2009 [8]		✓			✓				✓		✓	
	Vasquez et al. 2009 [55]		✓				?			✓			
	Kriegel et al. 2012 [52]	✓				✓				✓			
	Vasquez et al. 2014 [56]	✓				✓				✓			
	Vasquez et al. 2014 [53]	✓							✓	✓			
	Kriegel et al. 2015 [51]	✓				✓				✓			
	Vasquez et al. 2017 [54]		✓						✓	✓			

each iteration, the representation of the space evolves. Therefore, an effective stop criterion must be established. To this end, two main criteria can be highlighted. Pito [48,49], Vasquez et al. [53–56] and Loriot [8] performed a check for the minimum percentage of new surface scans at each iteration, whereas Kriegel et al. [51,52] checked the coverage rate of the holes in the mesh.

Similar to that in the SCP transposition methods, a different cost function was implemented for each method. The advantage of this type of method is that the criteria used can be easily eliminated or added.

5 Conclusion and future work

This article presents a state of the art review of the methods used to solve the view planning problem (VPP). The goal is to establish a scan plan to reconstruct or control a 3D object. The choice of the method clearly depends on both the type of object to be reconstructed and the technological means available to implement a solution. The types of available inputs directly define the type of method to be implemented. Overall, the methods can be divided into two groups: methods exploiting the *a priori* knowledge of the object to be scanned,

and methods that do not require such knowledge. Table 2 summarizes the evaluation techniques of these methods with respect to the technological criteria presented in Section 2.4, and Table 3 summarizes the positioning of the various methods with respect to the algorithmic criteria defined in Section 2.3. The advantages and disadvantages of each method were highlighted. The methods involving a transposition to the set covering problem (SCP) allow the definition of a scan plan with an input model, and constraints can be easily added or removed, as required. The methods that tend to optimize the coverage seek to cover the maximum surface, as specified in the input. The advantage of these methods is that they prioritize both the quality of the measurements and coverage of the model, and thus, accurate measurements can be made on the resulting point cloud. These methods are generally more industrialized than the other approaches. The advantage of search based methods is that the model to be scanned is not known. The view planning is therefore conducted iteratively in real time, and the point cloud can be later used to create a model of the scanned part.

Considering the limitations of the existing approaches, as highlighted in Tables 2 and 3, several perspectives can be emphasized. It can be noted that although machine learning is widely used in the field of computer graphics, none of the studied methods use this type of paradigm, likely because these methods require a large amount of data. Nevertheless, if a database were to exist in view planning, machine learning techniques can be likely applied to solve the VPP. Recently, some works based on this technology have been published by Mendoza et al. [58], Hepp et al. [59] and Devrim Kaba et al. [60].

Considering the time issue and the need to reduce the inspection time, nearly all the methods can be noted to postprocess the sensor positions to order them while solving the traveling salesman's problem. A future research direction can be to directly integrate this constraint in the algorithms, thereby removing the postprocessing step. Galceran et al. [61] and Cabreira et al. [62]

Table 3
Characterization of the approaches with respect to algorithmic criteria.

		Data discretization					General				Algorithm				Object				Sensor			
		Mesh (q)	Voxel grid (r)	Parametric surface (s)	Point cloud (t)	B-Rep (u)	View point generalization (a)	Object size generalization (b)	Support type considering (c)	Any type of constraint (d)	Self-terminating (e)	Scan minimization (f)	Covering maximization (g)	Time inspection minimization (h)	From CAD selection (i)	Minimal a priori knowledge (j)	Overlap (k)	Overlap analysis (l)	Occlusion detection (m)	Sensor quality measurement (n)	Collision detection (o)	Industrial part for validation (p)
SCP transposition	Scott et al. 2001 [10]	✓					✓	✓		✓	✓	✓	✓			✓			✓			
	Scott 2002 [9]	✓					✓	✓		✓	✓	✓	✓			✓	✓		✓			
	Martins et al. 2005 [19]		✓				~	✓		✓	✓	✓	✓			✓			✓	✓	✓	
	Scott 2009 [12]	✓					✓	✓		✓	✓	✓	✓			✓	✓		✓		✓	
	Loriot 2009 [8]	✓					✓	✓		✓	✓	✓	✓			✓			✓			
	Krause et al. 2011 [21]		~					✓		✓	✓	✓	✓		NA	✓	✓					NA
	Mahmud et al. 2011 [11]	✓					✓	✓		✓	?	✓	✓			✓			✓	✓		~
	Jing et al. 2016 [13]	✓	✓				✓	✓		✓	✓	✓	✓			✓			✓	✓		
	Hepp et al. 2017 [20]		✓				✓	✓		✓	✓	✓	✓	✓	NA	✓	✓		✓	✓	✓	NA
	Jing 2017 [14]	✓	✓				✓	✓		✓	✓	✓	✓			✓			✓	✓		
	Jing et al. 2018 [15]	✓	✓				✓	✓		✓	✓	✓	✓		NA	✓	✓		✓	✓		NA
	Jing et al. 2018 [16]	✓						~	✓	✓	✓	✓	✓	✓		✓			✓		✓	
	Mohammadikaji et al. 2018 [17]	✓						~	✓	✓	✓	✓	✓		✓	✓						✓
Covering optimization	Lee et al. 2000 [38]				✓		~	✓			✓		✓			✓	✓		✓	✓		
	Lee et al. 2001 [32]			~	✓		~	✓			✓		✓			✓	✓		✓			
	Son et al. 2002 [33]			~	✓		~	✓			✓		✓			✓	✓		✓			
	Souzani et al. 2006 [39]		✓				✓	✓			✓		✓			✓	?		✓	✓		NA
	Shi et al. 2007 [42]	✓					✓	✓		✓	✓		✓			✓				✓		✓
	Germani et al. 2009 [31]	✓		✓			✓	✓			✓		✓	✓		✓			✓			✓
	Germani et al. 2010 [43]			✓			✓	✓			~		✓	✓		✓				✓		✓
	Raffaeli et al. 2013 [35]	✓	✓	✓	✓		✓	✓			~		✓	✓		✓	✓	~	✓			✓
	Lartigue et al. 2014 [29]		✓				✓	✓			✓		✓			✓			✓	✓	✓	✓
	Bircher et al. 2015 [41]	✓					✓	✓			✓		✓	✓	NA	✓						NA
	Wu et al. 2015 [37]	✓					✓	✓			✓		✓			✓				✓		
	Ding et al. 2016 [40]	✓					~	?			✓		✓			✓			✓			✓
	Koutecky et al. 2016 [30]	✓	✓				✓	✓			✓		✓	✓		✓			✓		✓	✓
	Phan et al. 2018 [36]	✓		~				✓			✓		✓			✓	✓			✓		
	Sadaoui et al. 2018 [28]	✓						✓			?		✓		✓	✓	?		✓	✓	✓	✓
Search-based	Pito 1997–1999 [48,49]	✓									✓		✓		NA	NA	✓					
	Loriot 2009 [8]	✓					✓				✓		✓		NA	NA			✓			
	Vasquez et al. 2009 [55]		✓				✓			✓	✓		✓	✓	NA	NA	✓		✓	✓	✓	
	Kriegel et al. 2012 [52]	✓	✓				✓	✓		✓	✓		✓		NA	NA	✓		✓	✓	✓	
	Vasquez et al. 2014 [56]		✓				✓			✓	✓		✓	✓	NA	NA	✓		✓	✓	✓	
	Vasquez et al. 2014 [53]		✓					~	✓	✓	✓		✓	✓	NA	NA	✓		✓		✓	
	Kriegel et al. 2015 [51]	✓	✓				✓	✓		✓	✓		✓		NA	NA	✓		✓	✓	✓	
	Vasquez et al. 2017 [54]		✓					~	✓	✓	✓		✓	✓	NA	NA	✓		✓		✓	

propose reviews of methods solving the coverage path planning problem, which is a variant of the traveling salesman problem. The idea is to determine a path that passes through all points of an area or volume. This kind of methods could maybe also be studied and adapted to solve the view planning problem.

There are many papers published every day on this particular research topic. Indeed the problem being NP-complete, a unique solution do not exist and each applications can have its particular solution. We can mention the recent paper of Song et al. [63] for illustration.

Another constraint that has been highlighted in certain methods but never addressed is the quality of the overlap. When two scans are performed, they must be aligned. Depending on the alignment methods used, an overlap between the point clouds is necessary, although some constraints must be implemented on this overlap. For example, an overlap on a flat surface does not provide sufficient information to enable a correct alignment between the two scans, because the degree of freedom is extremely important to ensure proper registration. Thus, it can be considered that the study of the shapes contained in the overlapping areas can help further understand the remaining degrees of freedom to improve the quality of the alignment of the scans, and thereby, intrinsically, the quality of the final point cloud.

In the various works presented in this paper, the uncertainties of the measures are not addressed. Their estimation as well as the estimation of the point cloud quality could be of great interest in further optimizing the pose of the view points.

Moreover, as we see in Sections 3.1.6, 3.2.4 and 4.5, the validation and evaluation methods are completely different for each work, and a complete evaluation of the approaches using the same criteria is challenging. A research on fair evaluation metrics can be a subject of future works in this area.

Alignment methods can also be a considered in future work. Even though different registration methods exist, in some cases, classical methods such as the iterative closest point (ICP) cannot find a proper solution, as several solutions may exist. For example, asymmetric objects cannot be properly aligned using this type of method. Consequently, new methods must be implemented. For instance, the new methods could be implemented with a singularity in the sensor field or by using photogrammetry.

In conclusion, the capabilities of the examined methods and the foreseen prospects can help in better understanding and

resolving the numerous challenges associated with Industry 4.0. Indeed, the problem of view planning can be extended to many applications in addition to part manufacturing control: surveillance, building reconstruction, underwater inspection, site exploration, etc. This type of applications can be exploited with different varieties of robots not mentioned in this manuscript such as mobile robots AGV and AUV.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] Zhang S. High-speed 3D shape measurement with structured light methods: A review. Opt Lasers Eng 2018;106:119–31.

[2] Scott W, Roth G, Rivest J. View planning for automated three-dimensional object reconstruction and inspection. ACM Comput Surv 2003;35(1):64–96. http://dx.doi.org/10.1145/641865.641868, URL: http://doi.acm.org/10.1145/641865.641868.

[3] Zeng R, Yuhui W, Wang Z, Yong-Jin L. View planning in robot active vision: A survey of systems, algorithms, and applications. Comput Vis Media 2020;1–21, P..A.Y...p....

[4] Scott W, Roth G, Rivest J. View planning as a set covering problem. NRC Publ Arch 2001. http://dx.doi.org/10.4224/8913444.

[5] Karp R. Reducibility among combinatorial problems. In: Complexity of computer computations. Springer; 1972, p. 85–103.

[6] Piegl L, Tiller W. The NURBS book. Springer Science & Business Media; 2012.

[7] Stroud I, Nagy H. Solid modelling and CAD systems: How to survive a CAD system. Springer Science & Business Media; 2011.

[8] Loriot B. Automation of acquisition and post-processing for 3D digitalisation (Phd thesis), Université de Bourgogne; 2009, URL: https://tel.archives-ouvertes.fr/tel-00371269.

[9] Scott W. Performance-oriented view planning for automated object reconstruction (Phd thesis), University of Ottawa; 2002.

[10] Scott W, Roth G, Rivest J. View planning for multi-stage object reconstruction. 2001.

[11] Mahmud M, Joannic D, Roy M, Isheil A, Fontaine J. 3D part inspection path planning of a laser scanner with control on the uncertainty. Comput Aided Des 2011;43(4):345–55. http://dx.doi.org/10.1016/j.cad.2010.12.014, URL: http://www.sciencedirect.com/science/article/pii/S0010448510002472.

[12] Scott W. Model-based view planning. Mach Vis Appl 2009;20(1):47–69.

[13] Jing W, Polden J, Lin W, Shimada K. Sampling-based view planning for 3D visual coverage task with unmanned aerial vehicle. In: 2016 IEEE/RSJ international conference on intelligent robots and systems (IROS). 2016, p. 1808–15.

[14] Jing W. Coverage planning for robotic vision applications in complex 3D environment (Ph.D. thesis), Carnegie Mellon University; 2017.

[15] Wei J, Shimada K. Model-based view planning for building inspection and surveillance using voxel dilation, medial objects, and random-key genetic algorithm. J Comput Des Eng 2018;5(3):337–47. http://dx.doi.org/10.1016/j.jcde.2017.11.013, URL: http://www.sciencedirect.com/science/article/pii/S2288430017300866.

[16] Jing W, Goh CF, Rajaraman M, Gao F, Park S, Liu Y, Shimada K. A computational framework for automatic online path generation of robotic inspection tasks via coverage planning and reinforcement learning. IEEE Access 2018;6:54854–64. http://dx.doi.org/10.1109/ACCESS.2018.2872693.

[17] Mohammadikaji M, Bergmann S, Irgenfried S, Beyerer J, Dachsbacher C, Worn H. Inspection planning for optimized coverage of geometrically complex surfaces. In: 2018 workshop on metrology for industry 4.0 and IoT. 2018, p. 52–67. http://dx.doi.org/10.1109/METROI4.2018.8428313.

[18] Shimada K, Gossard D. Bubble mesh: Automated triangular meshing of non-manifold geometry by sphere packing. In: ACM sympossium on solid modeling and applications. ACM; 1995, p. 409–19.

[19] Martins F, Gómez García-Bermejo J, Zalama Casanova E, Perán González J. Automated 3D surface scanning based on CAD model. Mechatronics 2005;15(7):837–57. http://dx.doi.org/10.1016/j.mechatronics.2005.01.004, URL: http://www.sciencedirect.com/science/article/pii/S0957415805000371.

[20] Hepp B, Nießner M, Hilliges O. Plan3D: Viewpoint and trajectory optimization for aerial multi-view stereo reconstruction. 2017, CoRR abs/1705.09314. URL: http://arxiv.org/abs/1705.09314. arXiv:1705.09314.

[21] Krause A, Guestrin C, Gupta A, Kleinberg J. Robust sensor placements at informative and communication-efficient locations. ACM Trans Sensor Netw 2011;7(4):31.

[22] Tarbox G, Gottschlich S. Planning for complete sensor coverage in inspection. Comput Vis Image Underst 1995;61(1):84–111. http://dx.doi.org/10.1006/cviu.1995.1007, URL: http://www.sciencedirect.com/science/article/pii/S1077314285710077.

[23] Prieto F. Métrologie assistée par ordinateur. Apport des capteurs 3D sans contact (Phd thesis), Ecole de technologie supérieure; 2000.

[24] Papadimitriou CH, Steiglitz K. Combinatorial optimization: Algorithms and complexity. USA: Prentice-Hall, Inc.; 1982.

[25] Lan G, DePuy G, Whitehouse G. An effective and simple heuristic for the set covering problem. European J Oper Res 2007;176(3):1387–403. http://dx.doi.org/10.1016/j.ejor.2005.09.028, URL: http://www.sciencedirect.com/science/article/pii/S0377221705008313.

[26] DePuy G, Moraga R, Whitehouse G. Meta-RaPS: a simple and effective approach for solving the traveling salesman problem. Transp Res E 2005;41(2):115–30. http://dx.doi.org/10.1016/j.tre.2004.02.001, URL: http://www.sciencedirect.com/science/article/pii/S1366554504000146.

[27] Houck CR, Joines J, Kay MG. A genetic algorithm for function optimization: a Matlab implementation. Ncsu-Ie Tr 1995;95(09):1–10.

[28] Sadaoui S, Mehdi-Souzani C, Lartigue C. Computer-aided inspection planning: A multisensor high-level inspection planning strategy. J Comput Inf Sci Eng 2018;19. http://dx.doi.org/10.1115/1.4041970.

[29] Lartigue C, Quinsat Y, Mehdi-Souzani C, Zuquete Guarato A, Tabibian S. Voxel-based path planning for 3D scanning of mechanical parts. Comput-Aided Des Appl 2014;220–7, URL: https://hal.archives-ouvertes.fr/hal-01092685.

[30] Koutecky T, Palousek D, Brandejs J. Sensor planning system for fringe projection scanning of sheet metal parts. Measurement 2016;94. http://dx.doi.org/10.1016/j.measurement.2016.07.067.

[31] Germani M, Mengoni M, Raffaeli R. Automation of 3D view acquisition for geometric tolerances verification. In: 2009 IEEE 12th international conference on computer vision workshops, ICCV workshops. 2009, p. 1710–7.

[32] Lee K, Park H, Son S. A framework for laser scan planning of freeform surfaces. Int J Adv Manuf Technol 2001;17:171–80. http://dx.doi.org/10.1007/s001700170187.

[33] Son S, Park H, Lee K. Automated laser scanning system for reverse engineering and inspection. Int J Mach Tools Manuf 2002;42(8):889–97.

[34] Raffaeli R, Germani M, Mandorli F, Mengoni M. Computation of optimal acquisition viewpoints for the 3D optical inspection of mechanical components. In: Proceedings of the IMProVe 2011 international conference on innovative methods in product design. 2011.

[35] Raffaeli R, Mengoni M, Germani M. Context dependent automatic view planning: The inspection of mechanical components. Comput-Aided Des Appl 2013;10:111–27. http://dx.doi.org/10.3722/cadaps.2013.111-127.

[36] Phan N, Quinsat Y, Lavernhe S, Lartigue C. Scanner path planning with the control of overlap for part inspection with an industrial robot. Int J Adv Manuf Technol 2018;98(1):629–43. http://dx.doi.org/10.1007/s00170-018-2336-8.

[37] Wu Q, Lu J, Zou W, Xu D. Path planning for surface inspection on a robot-based scanning system. In: 2015 IEEE international conference on mechatronics and automation (ICMA). 2015, p. 2284–9. http://dx.doi.org/10.1109/ICMA.2015.7237842.

[38] Lee H, Park H. Automated inspection planning of free-form shape parts by laser scanning. Robot Comput-Integr Manuf - Robot Comput-Integr Manuf 2000;16:201–10. http://dx.doi.org/10.1016/S0736-5845(99)00060-5.

[39] Mehdi-Souzani C, Thiebaut F, Lartigue C. Scan planning strategy for a general digitized surface. J Comput Inf Sci Eng 2006. http://dx.doi.org/10.1115/1.2353853?, https://hal.archives-ouvertes.fr/hal-01226542.

[40] Ding L, Dai S, Mu P. CAD-based path planning for 3D laser scanning of complex surface. Procedia Comput Sci 2016;92:526–35. http://dx.doi.org/10.1016/j.procs.2016.07.378, URL: http://www.sciencedirect.com/science/article/pii/S1877050916316349. 2nd International Conference on Intelligent Computing, Communication & Convergence, ICCC 2016, 24-25 January 2016, Bhubaneswar, Odisha, India.

[41] Bircher A, Alexis K, Burri M, Oettershagen P, Omari S, Mantel T, Siegwart R. Structural inspection path planning via iterative viewpoint resampling with application to aerial robotics. In: 2015 IEEE international conference on robotics and automation (ICRA). IEEE; 2015, p. 6423–30.

[42] Shi Q, Xi N, Spagnuluo C. A feedback design to a CAD-guided area sensor planning system for automated 3D shape inspection. Comput-Aided Des Appl 2007;4(1–4):209–18. http://dx.doi.org/10.1080/16864360.2007.10738541.

[43] Germani M, Mandorli F, Mengoni M, Raffaeli R. CAD-based environment to bridge the gap between product design and tolerance control. Precis Eng 2010;34(1):7–15. http://dx.doi.org/10.1016/j.precisioneng.2008.10.002.
[44] Helsgaun K. An effective implementation of the Lin–Kernighan traveling salesman heuristic. European J Oper Res 2000;126(1):106–30. http://dx.doi.org/10.1016/S0377-2217(99)00284-2, URL: http://www.sciencedirect.com/science/article/pii/S0377221799002842.
[45] Cho M, Lee H, Yoon G, Choi J. A feature-based inspection planning system for coordinate measuring machines. Int J Adv Manuf Technol 2005;26(9):1078–87.
[46] Shintyakov D. Suboptimal travelling salesman problem (TSP) solver. 2017, https://github.com/dmishin/tsp-solver.
[47] Zuquete Guarato A. Métrologie 2D de pièces de forme complexes par moyens optiques : Une application à l'équilibrage ed vilbrequins. (Ph.D. thesis), Ecole normale supérieure de Cachan; 2013.
[48] Pito R. Automated surface acquisition using range cameras (Phd thesis), University of Pennsylvania; 1997.
[49] Pito R. A solution to the next best view problem for automated surface acquisition. IEEE Trans Pattern Anal Mach Intell 1999;21(10):1016–30. http://dx.doi.org/10.1109/34.799908.
[50] Yuan X. A mechanism of automatic 3D object modeling. IEEE Trans Pattern Anal Mach Intell 1995;17:307–11. http://dx.doi.org/10.1109/34.368196.
[51] Kriegel S, Rink C, Bodenmüller T, Suppa M. Efficient next-best-scan planning for autonomous 3D surface reconstruction of unknown objects. J Real-Time Image Process 2015;10(4):611–31.
[52] Kriegel S, Rink C, Bodenmüller T, Narr A, Suppa M, Hirzinger G. Next-best-scan planning for autonomous 3D modeling. In: 2012 IEEE/RSJ international conference on intelligent robots and systems. 2012, p. 2850–6.
[53] Vasquez-Gomez J, Sucar L, Murrieta-Cid R. View planning for 3D object reconstruction with a mobile manipulator robot. In: IEEE international conference on intelligent robots and systems. 2014, http://dx.doi.org/10.1109/IROS.2014.6943158.
[54] Vasquez-Gomez JI, Sucar LE, Murrieta-Cid R. View/state planning for three-dimensional object reconstruction under uncertainty. Auton Robots 2017;41(1):89–109. http://dx.doi.org/10.1007/s10514-015-9531-3.
[55] Vásquez-Gómez J, López-Damian E, Sucar L. View planning for 3D object reconstruction. In: Proceedings of the 2009 IEEE/RSJ international conference on intelligent robots and systems. IROS'09, Piscataway, NJ, USA: IEEE Press; 2009, p. 4015–20, URL: http://dl.acm.org/citation.cfm?id=1732643.1732707.
[56] Vasquez-Gomez J, Sucar L, Murrieta-Cid R, Lopez-Damian E. Volumetric next-best-view planning for 3D object reconstruction with positioning error. Int J Adv Robot Syst 2014;11(10):159. http://dx.doi.org/10.5772/58759.
[57] Vasquez-Gomez J, Sucar L, Murrieta-Cid R, Herrera-Lozada J. Tree-based search of the next best view/state for three-dimensional object reconstruction. Int J Adv Robot Syst 2018;15(1):1729881418754575. http://dx.doi.org/10.1177/1729881418754575.
[58] Mendoza M, Irving Vasquez-Gomez J, Taud H, Enrique Sucar L, Reta C. Supervised learning of the next-best-view for 3d object reconstruction. Pattern Recognit Lett 2020;133:224–31. http://dx.doi.org/10.1016/j.patrec.2020.02.024, URL: https://www.sciencedirect.com/science/article/pii/S0167865518305531.
[59] Hepp B, Dey D, Sinha SN, Kapoor A, Joshi N, Hilliges O. Learn-to-score: Efficient 3D scene exploration by predicting view utility. In: Proceedings of the European conference on computer vision (ECCV). 2018.
[60] Devrim Kaba M, Gokhan Uzunbas M, Nam Lim S. A reinforcement learning approach to the view planning problem. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR). 2017.
[61] Galceran E, Carreras M. A survey on coverage path planning for robotics. Robot Auton Syst 2013;61(12):1258–76. http://dx.doi.org/10.1016/j.robot.2013.09.004, URL: https://www.sciencedirect.com/science/article/pii/S092188901300167X.
[62] Cabreira TaM, Brisolara LB, Ferreira Jr. PR. Survey on coverage path planning with unmanned aerial vehicles. Drones 2019;3(1). http://dx.doi.org/10.3390/drones3010004, URL: https://www.mdpi.com/2504-446X/3/1/4.
[63] Song S, Kim D, Jo S. Online coverage and inspection planning for 3D modeling. Auton Robots 2020;44(8):1431–50.