Computational Visual Media
https://doi.org/10.1007/s41095-020-0179-3 Vol. 6, No. 3, September 2020, 225–245

Review Article

View planning in robot active vision: A survey of systems, algorithms, and applications

Rui Zeng[1], Yuhui Wen[1], Wang Zhao[1], and Yong-Jin Liu[1] (✉)

Abstract Rapid development of artificial intelligence motivates researchers to expand the capabilities of intelligent and autonomous robots. In many robotic applications, robots are required to make planning decisions based on perceptual information to achieve diverse goals in an efficient and effective way. The planning problem has been investigated in active robot vision, in which a robot analyzes its environment and its own state in order to move sensors to obtain more useful information under certain constraints. View planning, which aims to find the best view sequence for a sensor, is one of the most challenging issues in active robot vision. The quality and efficiency of view planning are critical for many robot systems and are influenced by the nature of their tasks, hardware conditions, scanning states, and planning strategies. In this paper, we first summarize some basic concepts of active robot vision, and then review representative work on systems, algorithms and applications from four perspectives: object reconstruction, scene reconstruction, object recognition, and pose estimation. Finally, some potential directions are outlined for future work.

Keywords robotic; view planning; active vision; next-best view; sensor planning

1 Introduction

Active robot vision [1] refers to the capability of a robot that can actively adjust its visual sensors to obtain useful information for various tasks. The related ideas of view planning [2–4], sensor planning [5–7], or next-best view (NBV) determination [8–10], play an important role in active vision. They enable robot vision systems to process and analyze current information to progressively cover or detect target objects (Fig. 1).

View planning can significantly improve the efficiency of robot systems [11–13]. Robots can perceive useful information from a single view. However, the information contained in a single view is limited due to the working range and field-of-view of each sensor. Furthermore, noise and errors are inevitable when converting analogue signals to digital data. Multiple views can provide additional information, and they can also filter information by averaging noise, resulting in more accurate data capture. Motivated by the idea of using multiple views, optimally planning a sequence of viewpoints for robots has been widely studied [11, 14, 15].

Due to the importance of view planning in active vision, many novel algorithms and applications have been proposed. This research has significantly advanced the perceptual ability of robot systems, and changed the behavior of robots in many areas [1, 16, 17], including services, medicine, industry, and agriculture. Vision-based tasks (object recognition

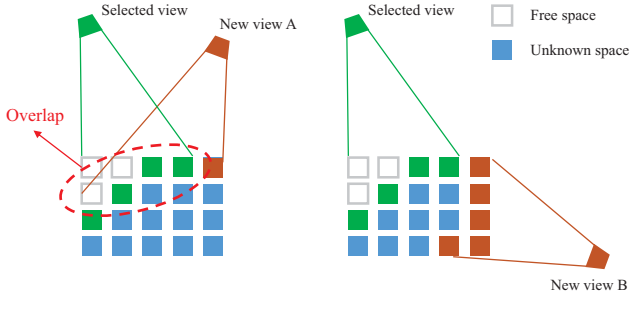

Fig. 1 Next-best view (NBV). Compared to new view A on the left, new view B on the right can collect more unknown information. It is thus more useful as the NBV.

1 BNRist, Department of Computer Science and Technology, Tsinghua University, Beijing, China. E-mail: R. Zeng, zengr17@mails.tsinghua.edu.cn; W. Zhao, zhao-w19@mails.tsinghua.edu.cn; Y.-J. Liu, liuyongjin@tsinghua.edu.cn (✉)

Manuscript received: 2020-03-18; accepted: 2020-05-01

and reconstruction, scene exploration, target tracking, object manipulation in rescue), where robots mainly rely on sequential visual information from sensors, benefit greatly from state-of-the-art view planning algorithms [17–19].

Although the view planning problem has been widely studied, there are still many unsolved challenges. For instance, inter-object occlusion increases the difficulty of data collection. Different settings of environmental lighting, surface materials, and textures also significantly affect data capture. Uncertainty about the environment surrounding the robot, variability of task requirements, imprecision of motion, and unreliability of visual perception are four key factors that hinder accurate perception for view planning [20, 21].

Despite these challenges, the development of active robot vision continues. There are many popular applications of active robot vision, including object reconstruction, scene reconstruction, object recognition, and pose estimation. In this paper, in addition to common characteristics shared by different view planning algorithms, we also summarize specific view planning algorithms designed for each of the above four applications.

This paper is organized as follows. First, we describe view planning systems and algorithms in Sections 2 and 3, respectively. Then, four applications of view planning in active robot vision: object reconstruction, scene reconstruction, object recognition, and pose estimation, are studied in Section 4. Finally, we suggest some directions for future work in Section 5 and offer concluding remarks in Section 6.

2 View planning system

The hardware system used in active vision tasks generally includes two kinds of components: robots and sensors[①]. The position and orientation of each robot determine the range of view of its sensors. The sensors themselves determine the type and quality of the collected data.

2.1 Robot

There are many kinds of robots [1, 21, 22], some of which are designed for specific purposes. This section introduces those kinds of robots commonly used in active vision tasks.

Robotic arms [14, 17] are used to move sensors to chosen positions within reach; the sensors take views in an environment containing one or more static objects that are generally close together. In some cases (e.g., as in Fig. 2), the base of the robotic arm is fixed. A sensor is attached to the end-effector of the robotic arm, whose cascade of joints determines the view of the sensor. Although the robotic arm has several degrees-of-freedom (DOFs) for collecting visual data, the fixed base limits the range of sensor movements. Such a robotic arm is applicable for tasks that do not require large movement of the robot.

Mobile robots are also widely used to move sensors. Compared to static robots, they are more flexible and suitable for active vision tasks in a large workspace, e.g., for reconstructing large objects or scenes (e.g., as in Fig. 3). As the sensors are generally mounted on the mobile robot, the problem of view planning for the robot becomes a problem of path planning. However, long-distance movement of mobile robots usually suffers from cumulative errors, and a large scene to be investigated generally has more uncertainties to be considered. As a result, researchers pay more attention to the flexibility, robustness, and accuracy of mobile robots.

2.2 Sensors

Different visual sensors have different characteristics suited to different scenarios. It is important to choose an appropriate sensor to cope with a specific vision problem. Various depth sensors (also called range sensors, range cameras, depth cameras, or RGB-D cameras) have been used in active vision systems to acquire 2.5D information about the observed target [2, 17, 20]. Depth sensors can be classified as passive or active, the latter using their own light sources for distance measurement.

Stereo vision, also called binocular stereo vision, is the most widely used measurement technology for passive sensors [2]. It obtains object depth information by imitating the human visual system [23]. In the passive measurement process, computing depth information relies on algorithms and is independent of the hardware. The disadvantages of passive measurements are twofold: they rely strongly on the target object or scene having texture, and they can suffer from degraded performance in the presence of lighting changes.

① As this paper focuses on visual sensors, by "sensor" we mean "visual sensor" if not otherwise qualified.

Fig. 2 Construction of a model of a plant using three robotic arms equipped with sensors. Reproduced with permission from Ref. [17], © IEEE 2019.

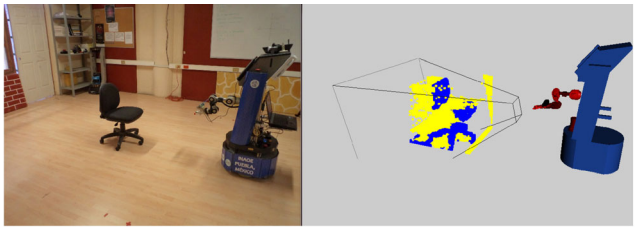

Fig. 3 A mobile robot equipped with a robotic arm, with a Kinect sensor mounted on the end-effector of the robotic arm. Reproduced with permission from Ref. [20], © Springer Nature 2015.

Time-of-Flight (ToF) and structured light are two commonly used measurement techniques for active sensors [2]. ToF sensors obtain distance information by measuring light travel time. The high-intensity and low-attenuation characteristics of lasers make ToF sensors suitable for long-to-medium distance measurements. A structured light sensor projects light with a specific structural pattern onto an object, and the result is then captured by another detector in the sensor. The computing unit calculates the depth value based on the reflected light. The advantages of structured light sensors are high quality and high frame rate. Thus, they are popular for measuring object surfaces in laboratories. Table 1 compares the three different types of depth sensor.

3 Data structures and algorithms for view planning

3.1 Data representations

View planning in an active vision system is based on detected information about an environment. The environment or workspace refers to the entire three-dimensional space which contains target objects (to be measured by sensors) and free space (for planning views) [2, 18, 24]. The active vision system updates its model of the environment using newly acquired information, which in turn guides view planning. Therefore, choice of data structure strongly affects the strategy of view planning [8, 24, 25]. In this subsection, we introduce some commonly used data representations (see Fig. 4).

3.1.1 Voxels

Voxel representation is popular in view planning because of its simplicity [24, 25]. It discretizes the environment using either an occupancy grid [2] or an octree [26]. Voxels can not precisely characterize fine details on the surface of a 3D object. However, the voxels can still represent the surface of complex objects to some extent, and the computational complexity of light/ray transmissions for view planning is acceptable in the discretized space. So, the voxel representation provides a good balance between quality and efficiency.

Fig. 4 Data representations. Left to right: triangle mesh, point cloud, voxel.

Table 1 Comparison of three types of depth sensor

Sensor type	Active	Distance	Accuracy	Affected by lighting	Depth measurements	Resolution	Cost
Stereo Vision	No	Short	High	Strongly	Sparse	High	Low
Time of Fight	Yes	Long	Low	Little	Dense	Low	High
Structured Light	Yes	Short	High	Little	Dense	Low	High

The principal disadvantage of the voxel representation is its wastefulness of storage. In a standard environment, the object to be measured only takes up a small portion of the whole environment, and the surface of the object occupies an even smaller portion of the voxelized space. Thus, voxel representation is more suitable for solid modeling than for surface modeling.

3.1.2 Triangle meshes

Triangle mesh representation is a common form of 3D model [27, 28]. Some view planning works [8, 29] use a triangle mesh to represent the intermediate state of the environment. A triangle mesh can well capture fine details on the surfaces of scanned objects. However, richer details may lead to higher computational costs. Furthermore, a triangle mesh only describes the scanned object's surface without considering space where no object exists. Therefore, applications of triangle mesh representation are less frequent than those using a voxel representation for view planning.

3.1.3 Point clouds

Point cloud is used for representing data collected by most of the existing depth sensors [30, 31]. Each point in the cloud contains not only spatial information (i.e., x, y, and z coordinates), but also reflection intensity and colour information. Compared to voxel representations, the point cloud data can restore more surface details of an object. In addition, the data is more straightforward and simple than triangle meshes. However, the point cloud data is similar to the triangle mesh representation in a way that is incapable of describing unknown and empty spaces. The point cloud and related representations (e.g., surfel representation [24]) have drawn lots of attention in the active vision.

3.2 Workflow

Fiugre 5 summarizes a typical active vision workflow, which includes view planning, motion planning, sensor scanning, and map updating. These four components are connected sequentially to form a closed loop. The actuator of a robot performs this closed-loop until a prescribed termination condition is satisfied. In this loop, the robot is initialized to a valid view by selecting a random view. The termination condition is based on objectives such as the scan range of the surface of the object, the uncertainty of object category recognition, the change of entropy in the workspace, etc.

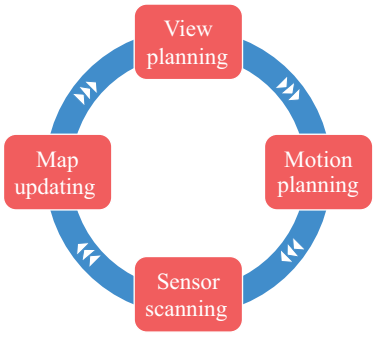

Fig. 5 Active vision workflow.

View planning is at the core of this workflow loop. After each round of information collection and updating, the view planner updates the NBV according to the current status and task goal.

3.3 Basic view planning algorithm

View planning aims to find a sequence of sensor views $\{v_i \mid i = 1, \ldots, n\} \subset V$ for an active vision task. $V = \mathbb{R}^3 \times SO(3)$ is the sensor view space. Researchers evaluate merit of a sequence of views in terms of running time and planning quality.

In fact, it is impractical to find the globally optimal solution $V^* \subset V$ for most active vision tasks. For example, in object reconstruction, using prior geometric knowledge about an object to find the smallest set of views to fully cover its surface has been proven to be an NP-complete problem [32]. Therefore, a view planning algorithm should find a near optimal sequence of views within an acceptable time.

Different vision tasks require different view planning algorithms, which can be broadly classified as search-based and synthesis-based approaches [33]. Most active vision work follows a search-based approach, also known as a generate-and-test approach [16, 17, 20, 27, 33–38]. These approaches generally sample a large number of candidate views, and then make view selections under specific constraints. The steps in a search-based approach may be summarized as follows:

- Sample a certain number of candidate views in the view space under some specific constraints (see Fig. 6).
- Perform a visual information simulation on each candidate view and estimate the amount of new information that can be gained from it. As well

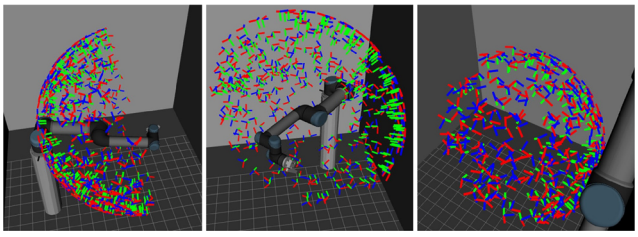

Fig. 6 Candidate views by sphere sampling. Reproduced with permission from Ref. [17], © IEEE 2019.

as information gain [25, 39], other factors can be considered such as the quality of the results [38, 40] and the cost of robot movement [14, 29].

- Choose the best viewpoint or set of viewpoints from these candidates.

The fundamental differences between view planning algorithms lie in the second and third steps. For example, in object *reconstruction* tasks, the main goal is added as much information as possible from each view [17, 25, 33, 41]. This criterion is used to select the candidate view. In object *recognition* tasks, algorithms typically select a view that minimizes uncertainty of which object is being viewed [16, 36, 37]. The system has to evaluate the critical information acquired.

Synthesis approaches use different approach to plan sensor views [6, 10, 41, 42]. With an analysis of task requirements, task constraints, system constraints, and sensor models, synthesis approaches calculate the NBV directly. As there is a process of simulating and evaluating a large number of views, the computational cost of the synthesis approach is much lower than that of the search-based approach. However, its performance in terms of accuracy and reliability is worse than for a search-based approach in the presence of complex occlusion and spatial uncertainty [41].

4 View planning by application

View planning algorithms use different data structures and strategies for different applications; different robots and sensors may also be involved. In this section, we summarize the characteristics of view planning algorithms targeting four applications: object reconstruction, scene reconstruction, object recognition, and pose estimation, and discuss some representative works to explain how view planning algorithms are applied to these specific tasks.

Object reconstruction aims to reconstruct the surface of a target object. It requires a robot-controlled sensor to scan the whole surface to build a three-dimensional (3D) model. Therefore, the view planner needs to enable the robot to capture as much information as possible in each view. Scene reconstruction requires mobile robots with sensors to traverse a scene, and comprehensively collect information about the entire scene. The purpose of object recognition is to use the sensor data from robots to identify the types of objects present. Unlike the above tasks, pose estimation may be regarded as an extension of object recognition. It attempts to accurately infer the poses of objects based on collected information by robots.

4.1 Object reconstruction

4.1.1 Overview

The aim of object reconstruction is to generate a 3D model of a physical object by sensing its surface from different viewpoints. Obtaining accurate 3D object models is required by many applications in industry, entertainment, culture, and architecture [2]. In these applications, a variety of physical objects such as cultural relics, houseware, or mechanical parts are scanned into digital counterparts and stored in the computer. The reconstruction process for an unknown object is generally performed in four subsequent steps: sensor positioning, sensing, registration, and view planning [2]. One of the challenging steps during reconstruction is view planning.

Traditional methods for object reconstruction manually operate sensors to scan objects from different views, so are slow and labour-intensive [43]. The rapid development of robotics and RGB-D (RGB and depth) sensors has led to more effective solutions for object reconstruction [30, 31]. A robot equipped with depth sensors can autonomously accomplish object reconstruction without manual control. After each image acquisition step, a view planner chooses the next-best view automatically using the current information. Then, the robot moves the sensors to the target view in a fully automated way [14, 17, 20]. Using active scanning, digital 3D geometric models can be reconstructed in the computer, as shown in Fig. 7. Compared to traditional manual approaches, scanning objects by autonomous robot systems with view planners achieves better reconstruction efficiency and accuracy.

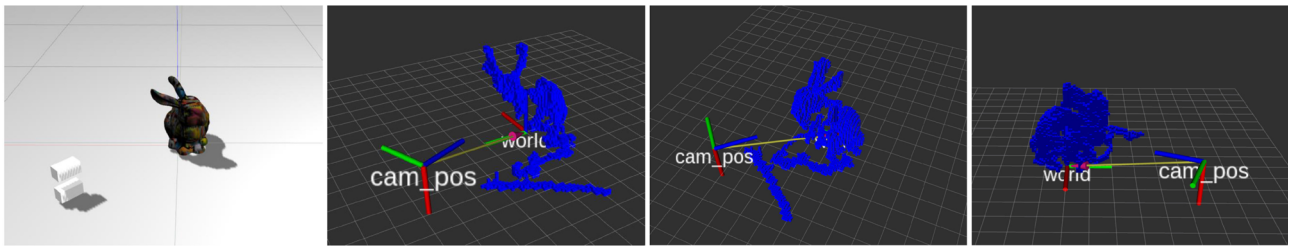

Fig. 7 Simulated active reconstruction. Left: camera and object model. Other images show the process of reconstruction.

4.1.2 Representative work

In the early stage of object reconstruction, view selection methods proposed by Connolly [41] and Wong et al. [42] followed the classic generate-and-test approach [6]. Both methods suggest sampling a set of candidate views using a voxel space. The number of unseen voxels revealed in each candidate view can be estimated and used as a criterion to find the NBV. Apart from such generate-and-test approaches, synthesis view planning methods have also been proposed [41, 42]. In these, the sensor viewpoints are directly determined from specific constraints modeled by analytical functions [6]. First, the observation direction of the NBV is calculated by accumulating the best observation direction of each unknown voxel. Then the exact position of the view is determined by spherical constraints. Both papers compare their synthesis approach with generate-and-test approaches, and show that, although fast in operation, the synthesis approach cannot handle occlusion problems and ensure availability of information in chosen views.

Instead of using a voxel representation [41, 42], active vision systems can also search for the NBV using a triangle mesh representation. In the algorithm proposed by Pito [8], the scanned surface of the object is recorded as the visible surface, and the core idea is to explore unknown parts connected to the visible surface to choose the NBV. The view planning algorithm can improve the efficiency of object reconstruction with registration constraints.

As object reconstruction proceeds, more and more object information is acquired. Using this observation, Banta et al. [10] presented three different view planning algorithms for use during different stages of scanning. The algorithms range from simple to complicated, and from coarse to fine, providing view planning that meets the requirements of different stages.

It is important to ensure not only the efficiency but also the quality of view planning for object reconstruction. Massios et al. [38] took the quality of reconstruction into account by defining the utility function as

$$f_{\text{total}}(v) = \omega_v f_{\text{visibility}}(v) + \omega_q f_{\text{quality}}(v) \quad (1)$$

where $f_{\text{visibility}}$ denotes the amount of information gain, which is calculated by counting the unknown boundary voxels of a view. f_{quality} denotes the reconstruction quality associated with voxels occupied by an object. It is defined as $\hat{n} \cdot \hat{v}$, where $\hat{n}$ is the local surface normal and $\hat{v}$ is the view direction. By defining the utility function in this way, the view planner tends to choose candidate views by considering both the amount of information and its quality.

Wu et al. [40] proposed a novel view planning algorithm intended for high-quality object reconstruction. After obtaining a local point cloud of the object, an approximate Poisson surface of the object is obtained using a weighted locally optimal projection operator [44] and screened Poisson surface reconstruction method [45, 46]. Poisson sampling [12] is then used to discretize the Poisson surface into a set of point clouds with associated directions. For each point in the point cloud, its confidence score is calculated according to completeness and smoothness to obtain a confidence heat map. Using it, visual evaluation is performed on each voxel in the voxel space to obtain the viewing vector field (VVF). This is used to select the best view for the next scan. Performance of the method is evaluated through physical experiments. Compared to a method based on exploration of boundaries delimiting unknown parts of the object's surface [11] and a method based on visual information [13], the proposed method is able to reconstruct a complete physical model faster, with fuller physical detail. Figure 8 shows the robot and reconstruction results from Ref. [40].

The idea of projecting rays in a simulation to assess has been applied in many works [11, 38, 40–42].

Fig. 8 Robot and modeling results. Reproduced with permission from Ref. [40], © ACM 2014.

Information gain metrics in these works can be classified as counting metrics, and probabilistic metrics [25]. Recently, Delmerico et al. [25] have suggested that candidate views can be ranked by information gain. In a voxel representation, the gain is defined as the volumetric information I contained in each voxel that is visible in a particular view. View planning is thus converted into a problem of choosing a view to maximize information gain.

Given a view v from a set of candidate views V, information gain is quantitatively evaluated by simulating projection lights R_v at the viewpoints according to the camera perspective model. Each ray $r \in R_v$ passes through many unknown voxels until known voxels or spatial boundaries block it, thereby forming a detected voxel set X. The information gain can be expressed as

$$G_v = \sum_{\forall r \in R_v} \sum_{\forall x \in X} I \tag{2}$$

where I is defined by the Shannon entropy formulation:

$$I = -P_o(x) \ln P_o(x) - \bar{P}_o(x) \ln \bar{P}_o(x) \tag{3}$$

Here, $P_o(x)$ is the probability of existence of the object in the voxel, while $\bar{P}_o(x) = 1 - P_o(x)$ is the probability that no object exists in the voxel. Equation (3) uses uncertainty about the voxel to measure the amount of unknown information. Five different volumetric information (VI) formulations can be derived from Eq. (3): occlusion aware VI, unobserved voxel VI, rear side voxel VI, rear side entropy VI, and proximity count VI (see Fig. 9), by introducing a Markov chain model or assigning different weights according to the spatial position of a voxel. These five VI formulations have been evaluated along with area factor VI [47] and average entropy VI [11] through simulation experiments. Results show that proximity count VI and area factor VI [47] are the best choices.

In most object reconstruction works, the object to be measured is placed on a desktop platform, so the sensor cannot see the bottom of the object. Krainin et al. [14] changed the view by lifting the object with a robotic arm. The view space can help to scan the object from all directions. In addition to selecting views based on information entropy, the method also considers the cost of movement of the robotic arm. Because of occlusion caused by the robotic arm, the grabbing posture must also be considered in the planning strategy. Candidate grabbing postures are generated using OpenRAVE [48] and evaluated to select the best posture. In addition to gripping objects with a robotic arm to extend the view space, there are also works that apply multiple types of sensors in a system, thereby collecting object information more flexibly. For example, tactile sensors and laser profilers can also be used to capture geometry and physical properties of objects. Cui et al. [29] proposed a multi-sensor based NBV framework. In each stage, the framework selects the next-best view from all executable views for all sensors. A geometric model is incrementally constructed by the framework through consecutive sensing actions. To obtain a high-quality detailed model, the terminatation criterion is that the largest

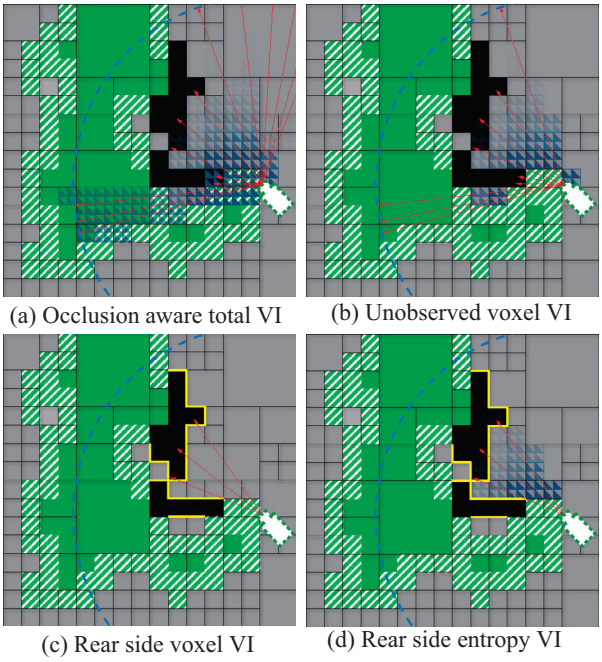

Fig. 9 Four VI formulations. Voxels colours are: black: occupied, grey: unknown, green: free, striped: frontier. Other markings are: yellow lines: object sides, red arrows: ray sets, dashed blue circle: maximal ray length blue triangles: VI weights. Proximity count VI behaves like rear side voxel VI (bottom left), but with a weight dependent on distance from previously observed surface voxels. Reproduced with permission from Ref. [25], © Springer Nature 2017.

triangle area in the mesh is below a threshold.

Deep learning approaches provide state-of-the-art performance in several areas, e.g., image classification [49]. However, studies on using learning methods for NBV prediction are rare [33]. Mendoza et al. [33] proposed a 3D convolutional neural network called NBV-Net which can effectively predict the NBV. Unlike the aforementioned works that perform a search to find the NBV [11, 25, 38, 40–42], NBV-Net directly predicts the NBV using a classification approach. Ground-truth NBV data are required to train the NBV-Net. The training data are collected based on the criterion that the NBV must maximize the scanned surface and also overlap the accumulated point cloud. The trained NBV-Net is capable of predicting the NBV to reconstruct several unknown objects. Furthermore, the NBV can be determined in a very short computation time because it avoids search.

View planning algorithms for object reconstruction may or may not be model-based [2]. Model-based algorithms use prior geometric knowledge about objects, while non-model-based algorithms can plan a sequence of views without considering the geometry of the object in advance. In practical applications, it is difficult to acquire prior knowledge before reconstructing 3D objects, so most existing view planning algorithms for object reconstruction are non-model based. However, in some applications, prior knowledge about objects may be used to improve the effectiveness of view planning algorithms. For example, prior knowledge of plant structures are have been utilized in a plant phenotyping task [17]. To help with NBV planning, a deep neural network is trained to predict plant-specific information in the form of a set of voxels. The environment is represented by two sets of voxels M and $\overline{M}$. M is the occupancy map built from information obtained by sensors. A point completion network [50] is used to predict the completion of the plant Y_p. Information from M and Y_p is combined to obtain $\overline{M}$, whose voxel set is currently unoccupied in M but is predicted to be occupied by Y_p. The NBV is determined as the candidate viewpoint that covers the most voxels in $\overline{M}$. Specifically, the approach casts rays from voxels in $\overline{M}$ to viewpoints to achieve more efficient view planning instead of the ray-casting strategy in traditional NBV planning.

Reinforcement learning (RL) has also been adopted to tackle the view planning problem, regarding it as a set covering optimization problem [51] with prior knowledge. Previous studies [52, 53] show that a greedy algorithm provides the best polynomial-time approximation algorithm to the NP-hard set-covering problem. A sequence of viewpoints for object reconstruction can be generated by solving a finite Markov decision process with reinforcement learning, which achieves better performance than greedy algorithms in almost all experimental cases. Specifically, each viewpoint is evaluated as following:

$$f_\lambda(X) = A(X)/L(X) \tag{4}$$

where X denotes the submesh obtained from selected viewpoints, and $A(X)$ denotes its total surface area, and $L(X)$ denotes the total boundary length of the area covered by X. $\lambda \in \mathbb{R}_{\geqslant 0}$ is determined by reinforcement learning; the viewpoint which maximizes $f_\lambda(X)$ is utilized. Subtle choice of λ by reinforcement learning is the key to making this algorithm's superiority to a greedy algorithm. When $\lambda = 0$, the algorithm is reduced to a greedy algorithm that uses area as the evaluation metric. When $\lambda > 0$, the algorithm also considers the shape of the submesh obtained. In some cases, selecting a viewpoint with a smaller area but a smoother surface will be more conducive to the overall reconstruction efficiency.

To summarize this section, view planning is widely used in object reconstruction, and typically works by ray casting in voxel space to evaluate the information gain of a view [11, 20, 25, 38, 41, 42]. Deep learning and reinforcement learning methods are beneficial for view planning [33, 51], but studies on these methods for reconstructing objects are rare. Machine learning methods offer a promising future direction for view planning for object reconstruction.

4.2 Scene reconstruction

4.2.1 Overview

3D scene models with fine details are vital in many fields [1, 54]: assisting engineers in analyzing building stability, preserving historical building information, planning pedestrian map navigation, etc. Manual sampling and modelling require a huge amount of time and effort for scene reconstruction, because of the large number of models in the scene. Entirely handing over the reconstruction task to an automatic mobile robot with sensors can greatly save manpower as well

as help to improve the quality of the reconstructed scene model.

Unlike object reconstruction, which observes one or more objects inside a limited volume, scene reconstruction requires obtaining a model of the entire 3D scene inside a target volume. Therefore, a robot used in scene reconstruction must be able to drive autonomously to each location in the scene. During the scanning process, the robot has to know its own position [55, 56]. The robot can then further plan its path and sensor views based on its position and an existing partial model of the scene, to allow it to scan the scene efficiently. The robot system chooses views to reconstruct the scene based on criteria such as information gain, robot movement cost, reconstruction quality, etc. [18, 35]. Compared with object reconstruction, a robot system for reconstructing a scene model must pay more attention to the cost of robot movement, which is a key factor in the efficiency of scene reconstruction.

4.2.2 Representative work

Among various types of scene reconstruction scenarios, indoor scene exploration and reconstruction have the most popular demand. With prior knowledge of scene geometry, i.e., a map of the building, indoor scene exploration becomes a classical 2D art gallery problem [57] (deciding how to arrange guards to watch an entire building, as shown in Fig. 10).

The art gallery problem can be reduced to a set-covering problem whose solution can be approximated by a greedy algorithm for 2D view planning with a robot [58]. The set-covering problem can be formulated as follows: for a finite set X, F is a subset of X's power set ($F \subset P(X)$), and $X = \cup_{s \in F} S$. The set C with the fewest elements that satisfies $C \subset F$ and $X = \cup_{s \in C} S$ is the best-view set choosen by view planning. Nüchter et al. [34] modeled the art gallery as a horizontal plane, and extended the existing approach [58] to address view

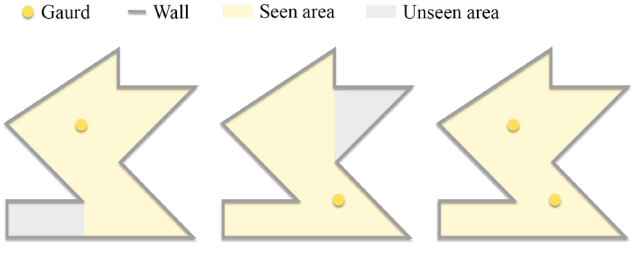

Fig. 10 Art gallery problem: for a polygonal gallery, how many guards are needed to cover the whole area?

planning with a robot for 3D scene reconstruction by considering several horizontal planes at different heights. NBVs are chosen by calculating the amount of 2D information that can be obtained in horizontal planes at a viewpoint.

While this method uses 2D data for view planning for 3D scene reconstruction [34], Blaer and Allen [35] provided ideas for view planning in large-scale indoor and outdoor sites with 3D data. The scanning process has two stages. In the first stage, an initial set of views is generated based on a 2D map. To take 3D scans at each view in the initial set, a path planning module [59] is combined with a robot. Then, an initialization model is constructed from the 3D scans for the second stage, in which a voxel-based occupancy procedure is adopted to plan NBVs.

In addition to using mobile robots that walk on the ground [34, 58, 59], an aerial robotic platform based on a micro aerial vehicle (MAV) can also be used for scene reconstruction [22]. Bircher et al. [22] proposed a view planning approach for the MAV to explore scene space. The approach samples views as nodes in a random tree using a tree construction algorithm such as RRT [60] or RRT-Star [61]. NBVs are determined by evaluating each branch in the tree for the amount of unknown information gain. In each iteration of view planning, the first edge of the best branch is executed to scan and update the scene. The MAV repeats the iteration process in a receding horizon fashion until exploration is complete. Simulation and physical experiments have been conducted to verify that sophisticated spaces can be dealt with in real time by the view planner on the MAV platform, which usually has limited computing resources.

Traditional active vision systems acquire 2D scene maps [34, 58] or 3D scanned data [59] to expand knowledge of the scene. Detailed object models in the scene are reconstructed from the acquired data by performing offline analysis. Xu et al. [62] proposed an autonomous system for scene reconstruction with online object analysis. The system is the first to integrate robot interaction with proactive validation for object extraction and scene segmentation, resulting in object-wise-quality reconstruction. Based on this autonomous system, Xu et al. [63] reconstructed the scene by online identification of objects using a 3D shape database. Object recognition is achieved by a novel recurrent

network with subnetworks for input processing, information aggregation, action generation, and next view prediction. The retrieved 3D objects are inserted into the scene to progressively replace the corresponding object scans for scene reconstruction.

Unlike methods which realize object extraction with the help of physical interaction [62], Liu et al. [19] proposed a novel active vision system, which provides object-aware guidance for on-the-fly scene exploration and object recognition. In one navigation pass, the proposed system first decides which object in the scene should be regarded as a target. Globally, the system uses multi-class graph cut minimization for segmentation to find candidate objects; the target object is selected based on database matching degree and robot movement cost. The robot then moves to the target object, and object-aware information gain is used to plan the NBV for local scanning. After the current target object has been recognized and reconstructed, the robot continues navigation by identifying and modeling the next target object. The robot sequentially visits and scans all objects in the scene to perform whole scene model reconstruction. Experiments show that the method is more accurate and efficient than most related work [19]. The authors have conducted simulation experiments based on the SUNCG [64] and ScanNet [65] scene data sets. Physical experiments were also conducted with a mobile robot carrying a Kinect sensor. Their experimental results verify that the system performs well in terms of reconstruction quality and efficiency; see Fig. 11. With a similar purpose, Zheng et al. [66] proposed a novel online reconstruction method with semantic segmentation for active understanding of unknown indoor scenes. The method adopts a volumetric representation, which is suitable for voxel labeling based on deep learning [27]. View planning is a view scoring field based not only information gain, but also safety, visibility, and movement cost. Then, the robot path and camera trajectory are jointly optimized for adjacent NBVs.

Recently, a multi-robot collaborative reconstruction system was proposed by Dong et al. [18]. Using multiple robots at the same time can greatly improve the speed of scene reconstruction, by minimizing the scanning effort of all robots while their collective coverage and reconstruction quality are maximized. In each iteration of view planning, the system selects the set of best views, in which each views is assigned to a robot. The planning procedure for assigning views can be formulated as a multiple traveling salesman problem (mTSP), which determines a path for each robot such that each task view is visited exactly once and the total traveling cost is minimized. However, mTSP is NP-hard [67]. Therefore, the authors propose a divide-and-conquer scheme to solve the problem in two steps. The first step assigns views to all robots by optimal mass transport (OMT), and then an optimal path for each robot is determined by a traveling salesman problem (TSP). OMT solves the view planning problem by an optimization process which takes movement cost and robot capacity into

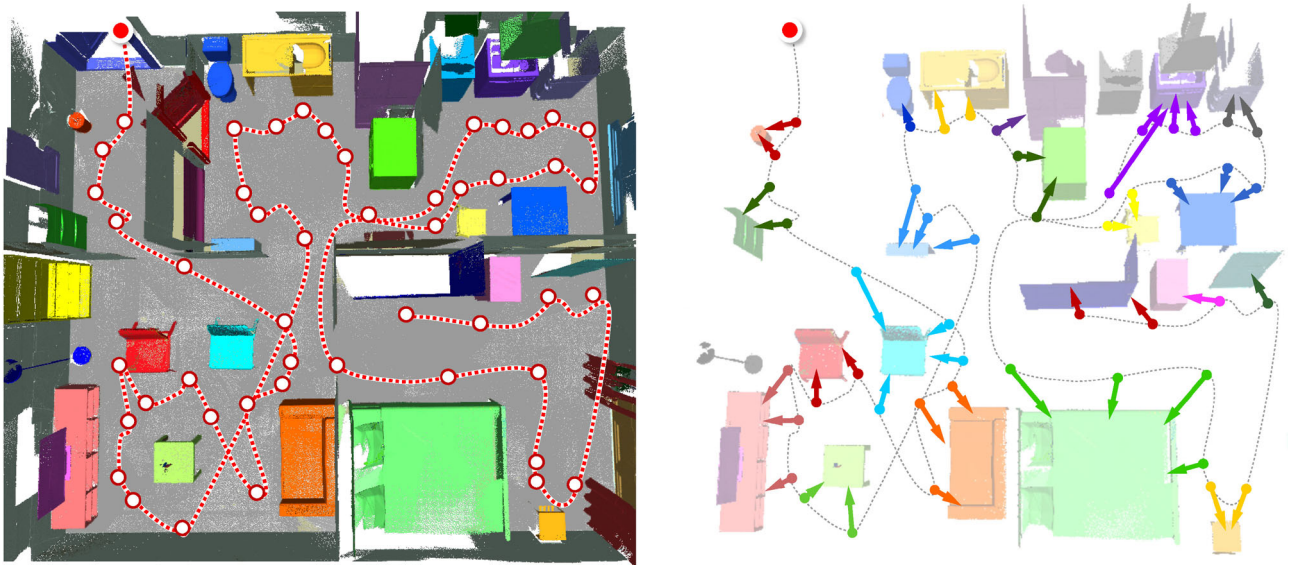

Fig. 11 Global and local view planning results. Reproduced with permission from Ref. [19], © ACM 2018.

consideration. The second step has to compute a TSP path, which is transformed into a smooth robot movement path. Each robot has to traverse the assigned views, as illustrated in Fig. 12.

With only a single depth image captured by a depth sensor as input, view planning can be used for high-quality scene reconstruction [68]. A deep reinforcement learning network DQN [69] is adopted for determining a sequence of viewpoints to complete the occlusion space in the depth image. The completion is done in an iterative fashion to make up all missing information. Each iteration chooses one viewpoint to render a new depth image, which is in-painted to fill the produced holes and re-projected into 3D space for the next iteration. In-painting is done by utilizing a 2D in-painting network [70] and SSCNet [64]. These steps are iterated until the complete scene point cloud is established. Compared with previous scene completion methods, SSCNet [64] and ScanComplete [71], the proposed method shows improved accuracy and completeness.

We end this section by illustrating the characteristics of scene reconstruction by a comparison with object reconstruction. Representative works in Sections 4.1 and 4.2 are summarized in Table 2. From the aspect of view planning algorithms, scene reconstruction and object reconstruction share great similarity. Both evaluate a view by the amount of information the view can acquire during the process of choosing NBVs. However, the main consideration in scene reconstruction is how to build a complete scene model efficiently with acceptable accuracy. Object reconstruction has a higher requirement for accuracy, needing greater object detail. On the other hand, scene reconstruction is more complicated than object reconstruction, as all object models in the entire area must be acquired. To achieve scene reconstruction, an active vision system has to process a large amount of information for view planning, during which uncertainties caused by collision and occlusion must be taken into account. Using aerial

Fig. 12 Multi-robot collaborative scene reconstruction algorithm. Reproduced with permission from Ref. [18], © ACM 2019.

Table 2 Representative reconstruction work

Task	Method	Representative
Object reconstruction	Predict the amount of unknown voxel observed by a view for evaluation.	Connoly [41]; Wong et al. [42]
Object reconstruction	Calculate the NBV direction by accumulating unknown information of voxel.	Connoly [41]; Wong et al. [42]
Object reconstruction	Explore unknown parts connected to the visible surface.	Pito [8]; Kriegel et al. [15]
Object reconstruction	Consider the reconstruction quality for NBV.	Massios and Fisher [38]; Wu et al. [40]
Object reconstruction	Fix sensors and change the pose of the object by a robotic arm to obtain new views.	Krainin et al. [14]
Object reconstruction	Select NBV from multiple types of sensors.	Cui et al. [29]
Object reconstruction	Use different view planning strategies in different states of scanning.	Banta et al. [10]
Scene reconstruction	Transform the 3D view planning problem into the 2D problem.	Nüchter et al. [34]
Scene reconstruction	Use boundary voxels between known and unknown volumes to select NBV.	Blaer and Allen [35]
Scene reconstruction	Research the balance between NBV and NBO.	Liu et al. [19]
Scene reconstruction	View planning for MAV.	Bircher et al. [22]
Scene reconstruction	View planning for Multi-robot collaboration system.	Dont et al. [18]

robots to explore large areas [22], object-guided scene reconstruction combined with object recognition [19], and multi-robot collaborative reconstruction [18] have all been proposed in recent studies, and are valuable and challenging issues for research.

4.3 Object recognition

4.3.1 Overview

Object recognition refers to identifying an object based on information provided by a sensor and a database. Recognizing objects from a single-view image has been well-studied, using e.g., template-based [72] or matching-based approaches [73, 74]. However, a number of practical issues (occlusion, lighting, texture) make single-view object recognition difficult.

Reasonably priced and widely used RGB-D sensors facilitate development of object recognition systems [75]. However, the problem of insufficient single-view information cannot be fully solved by improving hardware. Furthermore, noise in the captured data still has a great impact on the recognition accuracy when using a single view [37]. It is useful for sensors to acquire multiple features from different viewpoints.

View planning helps an object recognition system to complete recognition tasks with fewer views. While the reconstruction tasks mentioned in previous sections need to scan all information, the goal of view planning for object recognition is to locate critical object features to improve recognition accuracy and efficiency.

4.3.2 Representative work

In object recognition, information from the first view is often insufficient for confirming object type [27, 76]. Several assumptions about the types of objects are made based on a model database. Collecting further key information from additional viewpoints helps to eliminate ambiguity; this can be done until the object type can be uniquely determined [36, 77].

Hutchinson et al. [77] constructed a multi-sensor object recognition system based on the idea of reducing ambiguity. Using information from one of the sensors in the system, an initial set of hypotheses is formed. Then, the next sensor with an appropriate viewpoint is chosen to maximally disambiguate the initial set of hypotheses for object recognition. View planning in object recognition evaluates a candidate view based on the amount of ambiguity that would be resolved, as illustrated in Fig. 13.

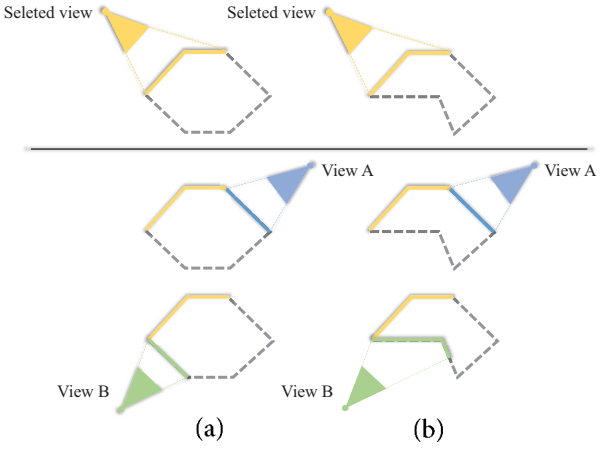

Fig. 13 Reducing ambiguity. Given hypotheses (a) and (b), view A cannot provide useful information, but view B can provide new information to verify which hypothesis is correct. Reproduced with permission from Ref. [77], © IEEE 1988.

Dickinson et al. [78] proposed an aspect hierarchy to assist in ambiguity judgement for view planning. The aspect hierarchy consists of aspects, faces, and boundary groups. The three levels of components hierarchically represent volumetric parts (from which each model in the database can be constructed), components of the aspects, and all subsets of contours bounding faces. With the help of the aspect hierarchy, objects are divided into image regions for inferring objects with more commonly used features such as lines or corners. If the system cannot uniquely determine the type of object, it will choose views based on a Bayesian network to reduce ambiguity. For recognizing objects composed of various basic shapes, the proposed method has excellent performance. However, the drawback of the system is that it cannot identify irregular manufactured objects.

Object recognition is also an ability required by a humanoid robot for its interaction with natural environments. Browatzki et al. [36] proposed a perception-driven multi-view recognition framework for interactive applications of a humanoid robot. An object is placed in the hands of the robot, and probabilistic Monte Carlo localization methods [79] are used to maintain multiple hypotheses about the object simultaneously. The robot rotates the object to select views that are beneficial for acquiring more critical information to resolve ambiguities and discriminate similar objects. Moreover, the system uses proprioceptive information to enhance the reliability of the hypotheses in addition to visual information. Using the iCub humanoid robot,

simulation experiments and physical experiments were carried out to demonstrate that the system can quickly and reliably confirm the type of the object. In particular, the humanoid robot can plan views to recognize objects with complex shape.

Kriegel et al. [75] constructed an active robot system integrating multiple tasks: scene exploration, object recognition, and reconstruction. Their robot system is equipped with a 3D camera and a laser stripe. The 3D camera is used to build an initial depth image for the whole scene quickly, while the high quality laser stripe scans and reconstructs objects. Each time the robot moves to a new position, it first uses the 3D camera to acquire a depth image, from which clusters are extracted using plane subtraction and Euclidean clustering. Then, the clusters are tested one by one to match models in the database. For a cluster that cannot be recognized, the system will plan new views to get more information for identifying or reconstructing it. If the system fails after several attempts to recognize the cluster, the recognition step is skipped, and the cluster is considered to be a new type of object, whose model is generated by the laser scanner and added to the object database.

Object recognition often requires manually designed features. Deep learning networks are widely used in adaptive and automatic methods, which can significantly overcome the shortcomings of manually designed features. Many works have introduced deep learning networks for automatically extracting features to represent objects. Wu et al. [27] established a convolutional deep belief network, ShapeNet, to learn and extract object features for the completion and recognition of objects. ShapeNet is capable of view synthesis from unseen viewpoints without manually labelling each image [80]. To train ShapeNet, a synthetic ModelNet dataset of 3D CAD meshes is constructed to provide sufficient data with images covering the full sphere of viewpoints around each object. If an object cannot be accurately identified by the network, the system can select the NBV actively to improve the recognition rate. The entropy of the classification distribution is

$$\begin{aligned} H &= H(p(y|x_o)) \\ &= -\sum_{k=1}^{K} p(y=k|x_o)\log p(y=k|x_o) \end{aligned} \tag{5}$$

where k is the tag for the category. According to each hypothesis of the network, given the new view V^i, $x_n^i = \text{Render}(x_u, x_o, V^i)$, where x_u is the unknown voxel, x_n^i is the newly observed voxel from V^i. To estimate the information entropy of a new view we compute:

$$\begin{aligned} H_i &= H(p(y|x_n^i, x_o)) \\ &= \sum_{x_n^i} p(x_n^i|x_o)H(y|x_n^i, x_o) \end{aligned} \tag{6}$$

The view V_i with the largest information entropy change $H - H_i$ is the best view for reducing ambiguity. The authors tested ShapeNet on the NYU real object dataset [81]. The results show that the proposed view planner using two view information for ten household objects outperforms other baseline methods with strategies including random selection, max visibility and furthest away.

In Ref. [27], a deep learning network is only used to assist in view evaluation. Johns et al. [76] proposed a method based on convolutional neural networks to infer the NBV for object recognition directly. The method proposed for recognition can be done over arbitrary camera trajectories with a classifier, which can be trained on image pairs in a sequence by weighting the contribution of each image pair. Thus, there is no need to train the networks over a potentially infinite number of camera paths. Conceptually, the method works on three levels. First, the authors trained a convolutional neural network (CNN-1); multiple images taken from different views are adopted as paired input to the network. For each pair of images, the network outputs a probability distribution which implicitly exhibits a measure of confidence based on entropy. The second level solves an NBV problem which investigates how to find the best-paired view. Finally, this work extends the NBV problem to the best view path problem, which is solved by constructing CNN-3 to evaluate the information gain of each particular view path and select the best view path accordingly. This method was shown to have better recognition accuracy than ModelNet [27] in an experimental comparison.

To summarize this section, the goal of object recognition is inherently different from object reconstruction. Object recognition focuses on key features that help to identify object types quickly and accurately, while more information has to be collected for complete object reconstruction. Thus, the planning strategies and evaluation criteria of

views for the two tasks are quite different. Reducing ambiguity is the key idea of view planning for object recognition. The algorithm often predicts a distribution of the object types by calculating uncertainties based on information entropy. The contribution of a new viewpoint is mainly determined by the reduction in ambiguity of object type. The most challenging problem in object reconstruction is to improve recognition accuracy in complex scenarios when objects occlude each other. Learning-based methods for automatically extracting features to match objects are worth considering [27, 76]. What's more, object recognition can be integrated with scene exploration in many practical applications [19, 75]. In future, more research will consider how to complete view planning in a multi-task system, and design strategies for missing models in databases.

4.4 Pose estimation

4.4.1 Overview

Pose estimation focuses on how to accurately locate objects in a scene using visual information, to enable further interaction with the objects. It is a popular topic in both industrial and computer-based tasks, including robot manipulation [82], augmented reality [83], etc.

Multiple views supply more clues for pose estimation; a single viewpoint is lacking in information. View planning is used to provide more critical information for matching objects in the scene with models in a database. This usage of view planning in pose estimation is similar to that in object recognition. However, pose estimation requires the system to not only match the object, but also to accurately recover its position and pose. Thus, the amount of information required in the pose estimation task is much more than for object recognition. Specifically, object recognition can use features from 2D images for matching without depth information, while depth information is required to accurately recover the pose of an object. Robot systems typically find key feature points that are helpful for calculating object pose with view planning algorithms.

4.4.2 Representative work

The information estimation and decision-making process in view planning is a typical Markov chain, and every view decision is a state transition. Eidenberg and Scharinger [16] proposed an effective partially observable Markov decision process for object recognition and pose estimation. The system establishes a high-dimensional Gaussian model for the object's pose and derives its state transition process. The system evaluates a view through entropy change from the probability distribution.

Wu et al. [84] simultaneously performed object recognition and pose estimation. In the initial stage, their system obtains initial hypotheses of objects and their relative poses for the scene model. The input RGB-D point cloud is divided into multiple clusters by the pre-processing method described in Ref. [85], and filtered to obtain candidate clusters that may contain objects. Then, feature descriptors (e.g., SURF [86] and SIFT [87]) of the clusters are extracted. With feature and consistency matching, the correspondences between the clusters and the database are determined. The system then uses the correspondences to estimate the pose of the candidate clusters by singular value decomposition (SVD) [88]. After generating candidate views of the current pose of the robot, a ray-casting simulation is performed on them to select the new view: that one that captures the most matching features. The process is repeated until the system can make a stable estimate.

Traditional machine learning methods play an important role in addressing the problem of pose estimation. Doumanoglou et al. [89] presented a novel framework using active random forests [90]. The framework is applied to solve the problem of view planning for classification, grab point detection, and pose estimation for the task of unfolding clothes by a robot. Other work proposed by Doumanoglou et al. [39] discusses the application of a Hough forest [91] to view planning for pose estimation. A Hough forest uses features automatically generated by an unsupervised auto-encoder, and then jointly performs object classification and pose recognition. For a new view, the information entropy is calculated based on the information stored in leaf nodes of the Hough forest. Reduction in entropy by a new view is used to select the NBV.

In pose estimation, mutual occlusion of objects brings obstacles to observation and feature extraction. Determining object poses in heavily occluded scenes is challenging. Sock et al. [37] established an active vision system for estimating poses of objects stacked in a highly crowded and cluttered environment. The

system first generates hypotheses based on state-of-the-art single object pose estimators. Then, the object hypotheses are used to predict the NBV. Often, the number of visible voxels is used for computing information entropy, but this measure is not suitable for 6D pose estimation for multiple objects in a crowded environment since it only considers coverage. Therefore, a viewpoint entropy that also considers saliency which can potentially reduce pose estimation uncertainty is proposed. In more detail, after each image acquisition step, the system uses latent class Hough forests (LHCF) [92] and a sparse autoencoder [93] to generate object pose hypotheses for views. The information obtained from every view is refined and subjected to registration correction processing. After the image acquisition and registration steps, the system uses an accumulated point cloud and multiple 6D object hypotheses to render candidate views for calculating information entropy. The view with the minimum view entropy is chosen as the NBV.

To summarize this section, pose estimation can be regarded as an extension of the object recognition task. There are many works that perform the two tasks in parallel [84]. However, because of different goals, the view planning process is different in the two tasks. In pose estimation, the system has multiple hypotheses about the scene model, describing which objects exist in the scene, and in what posture each object appears [37]. The view planner evaluates the reduction in uncertainty provided by new views based on the current multiple model hypotheses, and chooses the NBV [37, 39, 84]. In pose estimation, the robot system infers the types of objects in the scene and recovers their poses accurately at the same time. General pose estimation includes extracting object features, performing feature matching, and determining object poses. View planning has to select a suitable view to provide richer feature information, thereby completing the pose estimation process accurately and efficiently. In particular, machine learning methods such as random forests [90] and auto-encoders [93] are beneficial for view planning in the pose estimation task [37, 39, 89]. Estimating the poses of objects in a crowd is a challenging task worthy of future study.

5 Future trends

As view planning develops, the applications of view planning are becoming more practical and demanding. It is also noteworthy that more and more popular modern technologies, such as machine learning and deep neural networks, are combined into view planning. We outline the following potential directions for future works.

- With the rapid development of active vision, view planning tasks are becoming more practical. Research on this topic is not limited to laboratories; it has many practical applications in many industrial scenarios, such as manufacturing, home robot service, autonomous driving, etc.
- The continuous development of robots and sensors will have significant impact on view planning algorithms. For example, robots can provide a richer view space with more flexibility. In addition to colour and depth, haptic, temperature, and odour information provided by sensors may also be helpful for view planning.
- The capability of active vision systems to process multiple tasks at the same time deserves study in its own right. Recent works have increasingly tended to build more integrated robotic systems. For example, robotic systems that work on scene exploration often need to perform recognition and reconstruction of objects in the scene. Object recognition and pose estimation are inextricably linked. It is foreseeable that a better robot system will be able to process multiple tasks at the same time, and effective collaboration should bring benefits to every task.
- It is worth incorporating state-of-the-art algorithms from other related research areas, such as statistical mathematics, computer vision, computer graphics, and artificial intelligence into view planning. In particular, machine learning and deep learning technologies have attracted widespread attention. There are also successful applications of the technologies for feature extraction and analysis in view planning, as shown in Table 3.

6 Conclusions

In this paper, we have reviewed recent progress of view planning in active robot vision. The basic concepts of active vision and view planning were introduced, and then we thoroughly reviewed representative works targeting four goals: object reconstruction,

Table 3 View planning using traditional machine learning or deep learning

Ref.	Task	Technique	Description
[33]	Object reconstruction	3D Convolution Neural Network	Choose the NBV through 3D-CNN.
[51]	Object reconstruction	Reinforcement Learning	Use reinforcement learning to solve model-based view planning problem.
[17]	Object reconstruction	Point Cloud Network	Assist view evaluation by point cloud completion in plant phenotyping.
[89]	Pose estimation	Random Forest	Random forest assists the robot in unfolding clothes.
[39]	Pose estimation	Hough Forest, Sparse Autoencoder	Calculate the information entropy according to the distribution of leaf nodes in the Hough forest for evaluating the views.
[37]	Pose estimation	LCHF, Sparse Autoencoder	Pose estimation via LCHF or sparse Aatoencoder based on single-view.
[76]	Object recognition	Convolution Neural Network	Use CNN to evaluate the candidate views.
[27]	Object recognition	Convolutional Deep Belief Network	Assist view evaluation by network completion prediction.
[63]	Object recognition	Recurrent Neural Network	Simultaneously predict NBV and object category through RNN.
[68]	Scene reconstruction	Reinforcement Learning	Use DQN [69] to calculate NBV.

scene reconstruction, object recognition, and pose estimation. Although they share similarities in robotic active vision, many differences exist in the design of hardware and software. We show representative works by studying how to apply view planning algorithms in each application scenario. In summary, recent robotic view planning works have the following characteristics:

- Most studies follow search-based approaches, which generate candidate view sets based on task setting, system configuration, and algorithm constraints. Then, the next-best view is selected from these candidates; Table 4 compares works using search-based approaches.

Table 4 Work using search-based approaches

Ref.	Task	Data type	Sampling method	Utility function
[41]	Object reconstruction	Voxel	Sphere sample	Information gain
[42]	Object reconstruction	Voxel	Sphere sample	Information gain
[8]	Object reconstruction	Triangle mesh	Cylinder sample	Information gain
[38]	Object reconstruction	Voxel	Sphere sample	Information gain, quality
[40]	Object reconstruction	Voxel, point cloud	All voxels	Information gain, quality
[14]	Object reconstruction	Voxel	Sphere sample	Information gain, moving cost
[29]	Object reconstruction	Voxel, triangle mesh	No sample	Information gain, moving cost
[33]	Object reconstruction	Voxel	Sphere sample	Information gain
[51]	Object reconstruction	Triangle mesh	Around object	Information gain
[17]	Object reconstruction	Voxel	Sphere sample	Information gain
[34]	Scene reconstruction	Lines	Random sample	Information gain, moving cost
[35]	Scene reconstruction	Voxel	Voxels on ground	Information gain
[66]	Scene reconstruction	Voxel	Voxels on ground	Information gain, moving cost
[18]	Scene reconstruction	Voxel	Voxels on ground	Information gain, quality
[68]	Scene reconstruction	Voxel, point cloud	Sphere sample	Information gain
[19]	Object recognition	Voxel	Uniformly sample	Information gain
[77]	Object recognition	—	Sphere sample	Ambiguity
[78]	Object recognition	—	Sphere sample	Ambiguity
[36]	Object recognition	—	Sphere sample	Entropy
[75]	Object recognition	Voxel	Boundary search [94]	Information gain, quality
[27]	Object recognition	Voxel	Sphere sample	Entropy
[76]	Object recognition	—	Sphere sample	Entropy
[84]	Pose estimation	Point cloud, voxel	Neighbour sample	Information gain
[37]	Pose estimation	Point cloud	Sphere sample	Entropy
[16]	Pose estimation	—	Circularly sample	Entropy, moving cost
[89]	Pose estimation	—	Circularly sample	Entropy, moving cost

- View evaluation methods are mostly based on a greedy strategy, which prefers the maximum information gain or entropy change.
- To represent the environment, most works use voxels combined with a probabilistic statistical model.
- In reconstruction tasks, active vision systems usually explore unknown information in general. However, in object recognition or pose estimation tasks, systems tend to explore critical information about the object.
- In addition to information gain, a view planner also considers movement costs, the quality of current results, and registration accuracy.

With the continuous development of both robots and sensors, research on view planning in robot active vision will undoubtedly receive more and more attention. We have summarized potential trends that might inspire more researchers to propose valuable ideas and build more intelligent systems to better serve the future of human life.

Acknowledgements

The authors would like to thank Dr. Shihao Wu of Shenzhen VisuCA Key Laboratory/SIAT for providing Fig. 8, Dr. Vasquez-Gomez of the Instituto Nacional de AstrofÃŋsica ÃŞptica y ElectrÃşnica (INAOE) for providing Fig. 3, Prof. Kai Xu of the National University of Defense Technology and AICFVE Beijing Film Academy for providing Fig. 12, Dr. Xi Xia of the University of Science and Technology of China for providing Fig. 11, and Dr. Delmerico of the Robotics and Perception Group, University of Zurich for providing Fig. 9. This work was partially supported by a grant from the Science and Technology Department of Jiangsu Province, China.

References

[1] Chen, S. Y.; Li, Y. F.; Kwok, N. M. Active vision in robotic systems: A survey of recent developments. *The International Journal of Robotics Research* Vol. 30, No. 11, 1343–1377, 2011.

[2] Scott, W. R.; Roth, G.; Rivest, J. View planning for automated three-dimensional object reconstruction and inspection. *ACM Computing Surveys* Vol. 35, No. 1, 64–96, 2003.

[3] Roy, S. D.; Chaudhury, S.; Banerjee, S. Active recognition through next view planning: A survey. *Pattern Recognition* Vol. 37, No. 3, 429–446, 2004.

[4] Scott, W. R. Model-based view planning. *Machine Vision and Applications* Vol. 20, No. 1, 47–69, 2009.

[5] Tarabanis, K. A.; Tsai, R. Y.; Allen, P. K. Automated sensor planning for robotic vision tasks. In: Proceedings of the IEEE International Conference on Robotics and Automation, 76–82, 1991.

[6] Tarabanis, K. A.; Allen, P. K.; Tsai, R. Y. A survey of sensor planning in computer vision. *IEEE Transactions on Robotics and Automation* Vol. 11, No. 1, 86–104, 1995.

[7] Ye, Y. M.; Tsotsos, J. K. Sensor planning for 3D object search. *Computer Vision and Image Understanding* Vol. 73, No. 2, 145–168, 1999.

[8] Pito, R. A solution to the next best view problem for automated surface acquisition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 21, No. 10, 1016–1030, 1999.

[9] Pito, R. A sensor-based solution to the "next best view" problem. In: Proceedings of the 13th International Conference on Pattern Recognition, Vol. 1, 941–945, 1996.

[10] Banta, J. E.; Wong, L. R.; Dumont, C.; Abidi, M. A. A next-best-view system for autonomous 3-D object reconstruction. *IEEE Transactions on Systems, Man, and Cybernetics — Part A: Systems and Humans* Vol. 30, No. 5, 589–598, 2000.

[11] Kriegel, S.; Rink, C.; Bodenmüller, T.; Suppa, M. Efficient next-best-scan planning for autonomous 3D surface reconstruction of unknown objects. *Journal of Real-Time Image Processing* Vol. 10, No. 4, 611–631, 2015.

[12] Corsini, M.; Cignoni, P.; Scopigno, R. Efficient and flexible sampling with blue noise properties of triangular meshes. *IEEE Transactions on Visualization and Computer Graphics* Vol. 18, No. 6, 914–924, 2012.

[13] Khalfaoui, S.; Seulin, R.; Fougerolle, Y.; Fofi, D. An efficient method for fully automatic 3D digitization of unknown objects. *Computers in Industry* Vol. 64, No. 9, 1152–1160, 2013.

[14] Krainin, M.; Curless, B.; Fox, D. Autonomous generation of complete 3D object models using next best view manipulation planning. In: Proceedings of the IEEE International Conference on Robotics and Automation, 5031–5037, 2011.

[15] Kriegel, S; Rink, C; Bodenmüller, T; Narr, A; Suppa, M; Hirzinger, G. Next-best-scan planning for autonomous 3D modeling. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 2850–2856, 2012.

[16] Eidenberger, R; Scharinger, J. Active perception and scene modeling by planning with probabilistic 6D object poses. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 1036–1043, 2010.

[17] Wu, C. M.; Zeng, R.; Pan, J.; Wang, C. C. L.; Liu, Y. J. Plant phenotyping by deep-learning-based planner for multi-robots. *IEEE Robotics and Automation Letters* Vol. 4, No. 4, 3113–3120, 2019.

[18] Dong, S. Y.; Xu, K.; Zhou, Q.; Tagliasacchi, A.; Xin, S. Q.; Nießner, M.; Chen, B. Multi-robot collaborative dense scene reconstruction. *ACM Transactions on Graphics* Vol. 38, No. 4, Article No. 84, 2019.

[19] Liu, L.; Xia, X.; Sun, H.; Shen, Q.; Xu, J.; Chen, B.; Huang, H; Xu, K. Object-aware guidance for autonomous scene reconstruction. *ACM Transactions on Graphics* Vol. 37, No. 4, Article No. 104, 2018.

[20] Vasquez-Gomez, J. I.; Sucar, L. E.; Murrieta-Cid, R. View/state planning for three-dimensional object reconstruction under uncertainty. *Autonomous Robots* Vol. 41, No. 1, 89–109, 2017.

[21] Palomeras, N.; Hurtos, N.; Vidal, E.; Carreras, M. Autonomous exploration of complex underwater environments using a probabilistic next-best-view planner. *IEEE Robotics and Automation Letters* Vol. 4, No. 2, 1619–1625, 2019.

[22] Bircher, A; Kamel, M; Alexis, K; Oleynikova, H.; Siegwart, R. Receding horizon "next-best-view" planner for 3D exploration. In: Proceedings of the IEEE International Conference on Robotics and Automation, 1462–1468, 2016.

[23] Marr, D.; Poggio, T. A computational theory of human stereo vision. *Proceedings of the Royal Society B: Biological Sciences* Vol. 204, No. 1156, 301–328, 1979.

[24] Monica, R.; Aleotti, J. Surfel-based next best view planning. *IEEE Robotics and Automation Letters* Vol. 3, No. 4, 3324–3331, 2018.

[25] Delmerico, J.; Isler, S.; Sabzevari, R.; Scaramuzza, D. A comparison of volumetric information gain metrics for active 3D object reconstruction. *Autonomous Robots* Vol. 42, No. 2, 197–208, 2018.

[26] Hornung, A.; Wurm, K. M.; Bennewitz, M.; Stachniss, C.; Burgard, W. OctoMap: An efficient probabilistic 3D mapping framework based on octrees. *Autonomous Robots* Vol. 34, No. 3, 189–206, 2013.

[27] Wu, Z.; Song, S.; Khosla, A.; Yu, F.; Zhang, L.; Tang, X.; Xiao, J. 3D shapenets: A deep representation for volumetric shapes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1912–1920, 2015.

[28] Chang, A. X.; Funkhouser, T.; Guibas, L.; Hanrahan, P.; Huang, Q. X.; Li, Z. M.; Savarese, S.; Savva, M.; Song, S. R.; Su, H. et al. ShapeNet: An information-rich 3D model repository. *arXiv preprint* arXiv:1512.03012, 2015.

[29] Cui, J.; Wen, J. T.; Trinkle, J. A multi-sensor next-best-view framework for geometric model-based robotics applications. In: Proceedings of the International Conference on Robotics and Automation, 8769–8775, 2019.

[30] Zhang, Z. Y. Microsoft kinect sensor and its effect. *IEEE Multimedia* Vol. 19, No. 2, 4–10, 2012.

[31] Keselman, L.; Woodfill, J. I.; Grunnet-Jepsen, A.; Bhowmik, A. Intel realsense stereoscopic depth cameras. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1–10, 2017.

[32] Tarbox, G. H.; Gottschlich, S. N. Planning for complete sensor coverage in inspection. *Computer Vision and Image Understanding* Vol. 61, No. 1, 84–111, 1995.

[33] Mendoza, M.; Vasquez-Gomez, J. I.; Taud, H.; Sucar, L. E.; Reta, C. Supervised learning of the next-best-view for 3D object reconstruction. *Pattern Recognition Letters* Vol. 133, 224–231, 2020.

[34] Nüchter, A; Surmann, H.; Hertzberg, J. Planning robot motion for 3D digitalization of indoor environments. In: Proceedings of the 11th International Conference on Advanced Robotics, 78, 2003.

[35] Blaer, P. S.; Allen, P. K. Data acquisition and view planning for 3-D modeling tasks. In: Proceedings of the IEEE International Conference on Intelligent Robots and Systems, 417–422, 2007.

[36] Browatzki, B.; Tikhanoff, V.; Metta, G.; Bu¨lthoff, H. H.; Wallraven, C. Active object recognition on a humanoid robot. In: Proceedings of the IEEE International Conference on Robotics and Automation, 2021–2028, 2012.

[37] Sock, J.; Kasaei, S. H.; Lopes, L. S.; Kim, T. K. Multi-view 6D object pose estimation and camera motion planning using RGBD images. In: Proceedings of the IEEE International Conference on Computer Vision, 2228–2235, 2017.

[38] Massios, N. A.; Fisher, R. B. A best next view selection algorithm incorporating a quality criterion. In: Proceedings of the British Machine Vision Conference, 780–789, 1998.

[39] Doumanoglou, A.; Kouskouridas, R.; Malassiotis, S.; Kim, T. K. Recovering 6D object pose and predicting next-best-view in the crowd. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 3583–3592, 2016.

[40] Wu, S. H.; Sun, W.; Long, P. X.; Huang, H.; Cohen-Or, D.; Gong, M. L.; Deussen, O.; Chen, B. Q. Quality-driven poisson-guided autoscanning. *ACM Transactions on Graphics* Vol. 33, No. 6, Article No. 203, 2014.

[41] Connolly, C. The determination of next best views. In: Proceedings of the IEEE International Conference on Robotics and Automation, 432–435, 1985.

[42] Wong, L. M.; Dumont, C.; Abidi, M. A. Next best view system in a 3d object modeling task. In: Proceedings of the IEEE International Symposium on Computational Intelligence in Robotics and Automation, 306–311, 1999.

[43] Liu, Y. J.; Zhang, J. B.; Hou, J. C.; Ren, J. C.; Tang, W. Q. Cylinder detection in large-scale point cloud of pipeline plant. *IEEE Transactions on Visualization and Computer Graphics* Vol. 19, No. 10, 1700–1707, 2013.

[44] Huang, H.; Li, D.; Zhang, H.; Ascher, U.; Cohen-Or, D. Consolidation of unorganized point clouds for surface reconstruction. *ACM Transactions on Graphics* Vol. 28, No. 5, Article No. 176, 2009.

[45] Kazhdan, M.; Bolitho, M.; Hoppe, H. Poisson surface reconstruction. In: Proceedings of the 4th Eurographics Symposium on Geometry Processing, Vol. 7, 2006.

[46] Kazhdan, M.; Hoppe, H. Screened poisson surface reconstruction. *ACM Transactions on Graphics* Vol. 32, No. 3, Article No. 29, 2013.

[47] Vasquez-Gomez, J. I.; Sucar, L. E.; Murrieta-Cid, R.; Lopez-Damian, E. Volumetric next-best-view planning for 3D object reconstruction with positioning error. *International Journal of Advanced Robotic Systems* Vol. 11, No. 10, 159, 2014.

[48] Diankov, R.; Kuffner, J. OpenRAVE: A planning architecture for autonomous robotics. Technical Report CMU-RI-TR-08-34. Robotics Institute, Carnegie Mellon University, 2008.

[49] Krizhevsky, A.; Sutskever, I.; Hinton, G. E. ImageNet classification with deep convolutional neural networks. In: Proceedings of the 25th International Conference on Neural Information Processing Systems, Vol. 1, 1097–1105, 2012.

[50] Yuan, W.; Khot, T.; Held, D.; Mertz, C.; Hebert, M. PCN: point completion network. In: Proceedings of the International Conference on 3D Vision, 728–737, 2018.

[51] Kaba, M. D.; Uzunbas, M. G.; Lim, S. A reinforcement learning approach to the view planning problem. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 5094–5102, 2017.

[52] Dinur, I.; Steurer, D. Analytical approach to parallel repetition. In: Proceedings of the 46th Annual ACM Symposium on Theory of Computing, 624–633, 2014.

[53] Feige, U. A threshold of ln n for approximating set cover. *Journal of the ACM* Vol. 45, No. 4, 634–652, 1998.

[54] Smith, N.; Moehrle, N.; Goesele, M.; Heidrich, W. Aerial path planning for urban scene reconstruction: a continuous optimization method and benchmark. *ACM Transactions on Graphics* Vol. 37, No. 6, Article No. 183, 2019.

[55] Durrant-Whyte, H.; Bailey, T. Simultaneous localization and mapping: Part I. *IEEE Robotics & Automation Magazine* Vol. 13, No. 2, 99–110, 2006.

[56] Bailey, T.; Durrant-Whyte, H. Simultaneous localization and mapping (SLAM): Part II. *IEEE Robotics & Automation Magazine* Vol. 13, No. 3, 108–117, 2006.

[57] O'rourke, J. *Art Gallery Theorems and Algorithms, Vol. 57*. Oxford University Press, 1987.

[58] Gonzalez-Banos, H.; Mao, E; Latombe, J. C.; Murali, T. M.; Efrat, A. Planning robot motion strategies for efficient model construction. In: *Robotics Research*. Hollerbach, J. M.; Koditschek, D. E. Eds. Springer London, 345–352, 2000.

[59] Blaer, P.; Allen, P. K. Topbot: automated network topology detection with a mobile robot. In: Proceedings of the IEEE International Conference on Robotics and Automation, Vol. 2, 1582–1587, 2003.

[60] LaValle, S. M. Rapidly-exploring random trees: A new tool for path planning. 1998.

[61] Karaman, S.; Frazzoli, E. Sampling-based algorithms for optimal motion planning. *The International Journal of Robotics Research* Vol. 30, No. 7, 846–894, 2011.

[62] Xu, K.; Huang, H.; Shi, Y.; Li, H.; Long, P.; Caichen, J.; Sun, W.; Chen, B. Autoscanning for coupled scene reconstruction and proactive object analysis. *ACM Transactions on Graphics* Vol. 34, No. 6, Article No. 177, 2015.

[63] Xu, K.; Shi, Y.; Zheng, L.; Zhang, J.; Liu, M.; Huang, H.; Su, H.; Cohen-Or, D.; Chen. B. 3D attention-driven depth acquisition for object identification. *ACM Transactions on Graphics* Vol. 35, No. 6, Article No. 238, 2016.

[64] Song, S.; Yu, F.; Zeng, A.; Chang, A. X.; Savva, M.; Funkhouser. T. Semantic scene completion from a single depth image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 190–198, 2017.

[65] Dai, A.; Chang, A. X.; Savva, M.; Halber, M.; Funkhouser, T.; Nießner, M. ScanNet: Richly-annotated 3D reconstructions of indoor scenes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2432–2443, 2017.

[66] Zheng, L. T.; Zhu, C. Y.; Zhang, J. Z.; Zhao, H.; Huang, H.; Niessner, M.; Xu, K. Active scene understanding via online semantic reconstruction. *Computer Graphics Forum* Vol. 38, No. 7, 103–114, 2019.

[67] Bektas, T. The multiple traveling salesman problem: An overview of formulations and solution procedures. *Omega* Vol. 34, No. 3, 209–219, 2006.

[68] Han, X.; Zhang, Z.; Du, D.; Yang, M.; Yu, J.; Pan, P.; Yang, X.; Liu, L.; Xiong, Z.; Cui, S. Deep reinforcement learning of volume-guided progressive view inpainting for 3D point scene completion from a single depth image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 234–243, 2019.

[69] Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A. A.; Veness, J.; Bellemare, M. G.; Graves, A.; Riedmiller, M.; Fidjeland, A. K.; Ostrovski, G. Human-level control through deep reinforcement learning. *Nature* Vol. 518, No. 7540, 529–533, 2015.

[70] Liu, G. L.; Reda, F. A.; Shih, K. J.; Wang, T. C.; Tao, A.; Catanzaro, B. Image inpainting for irregular holes using partial convolutions. In: Proceedings of the European Conference on Computer Vision, 89–105, 2018.

[71] Dai, A.; Ritchie, D.; Bokeloh, M.; Reed, S.; Sturm, J.; Nießner, M. Scancomplete: Large-scale scene completion and semantic segmentation for 3D scans. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 4578–4587, 2018.

[72] Hinterstoisser, S.; Holzer, S.; Cagniart, C.; Ilic, S.; Konolige, K.; Navab, N.; Lepetit, V. Multimodal templates for real-time detection of texture-less objects in heavily cluttered scenes. In: Proceedings of the IEEE International Conference on Computer Vision, 858–865, 2011.

[73] Martinez, M.; Collet, A.; Srinivasa, S. S. Moped: A scalable and low latency object recognition and pose estimation system. In: Proceedings of the IEEE International Conference on Robotics and Automation, 2043–2049, 2010.

[74] Tang, J.; Miller, S.; Singh, A.; Abbeel, P. A textured object recognition pipeline for color and depth image data. In: Proceedings of the IEEE International Conference on Robotics and Automation, 3467–3474, 2012.

[75] Kriegel, S.; Brucker, M.; Marton, Z.-C.; Bodenmüller, T.; Suppa, M. Combining object modeling and recognition for active scene exploration In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 2384–2391, 2013.

[76] Johns, E.; Leutenegger, S.; Davison, A. J. Pairwise decomposition of image sequences for active multi-view recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 3813–3822, 2016.

[77] Hutchinson, S. A.; Kak, A. C. Planning sensing strategies in a robot work cell with multi-sensor capabilities. *IEEE Transactions on Robotics and Automation* Vol. 5, No. 6, 765–783, 1989.

[78] Dickinson, S. J.; Christensen, H. I.; Tsotsos, J. K.; Olofsson, G. Active object recognition integrating attention and viewpoint control. *Computer Vision and Image Understanding* Vol. 67, No. 3, 239–260, 1997.

[79] Fox, D.; Burgard, W.; Dellaert, F.; Thrun, S; Monte Carlo localization: Efficient position estimation for mobile robots. In: Proceedings of the 16th National Conference on Artificial Intelligence and the 11th Innovative Applications of Artificial Intelligence Conference Innovative Applications of Artificial Intelligence, 343–349, 1999.

[80] Johns, E.; Mac Aodha, O.; Brostow, G. J. Becoming the expert-interactive multi-class machine teaching. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2616–2624, 2015.

[81] Silberman, N.; Hoiem, D.; Kohli, P.; Fergus, R. Indoor segmentation and support inference from RGBD images. In: Proceedings of the European Conference on Computer Vision, 746–760, 2012.

[82] Kouskouridas, R.; Charalampous, K.; Gasteratos, A. Sparse pose manifolds. *Autonomous Robots* Vol. 37, No. 2, 191–207, 2014.

[83] Makris, S.; Karagiannis, P.; Koukas, S.; Matthaiakis, A. S. Augmented reality system for operator support in human–robot collaborative assembly. *CIRP Annals* Vol. 65, No. 1, 61–64, 2016.

[84] Wu, K.; Ranasinghe, R.; Dissanayake, G. Active recognition and pose estimation of household objects in clutter. In: Proceedings of the IEEE International Conference on Robotics and Automation, 4230–4237, 2015.

[85] Richtsfeld, A.; Mörwald, T.; Prankl, J.; Zillich, M.; Vincze. M; Segmentation of unknown objects in indoor environments. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 4791–4796, 2012.

[86] Bay, H.; Ess, A.; Tuytelaars, T.; van Gool, L. Speeded-up robust features (SURF). *Computer Vision and Image Understanding* Vol. 110, No. 3, 346–359, 2008.

[87] Lowe, D. G. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision* Vol. 60, No. 2, 91–110, 2004.

[88] Arun, K. S.; Huang, T. S.; Blostein, S. D. Least-squares fitting of two 3-D point sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 9, No. 5, 698–700, 1987.

[89] Doumanoglou, A.; Kim, T. K.; Zhao, X. W.; Malassiotis, S. Active random forests: An application to autonomous unfolding of clothes. In: Proceedings of the European Conference on Computer Vision, 644–658, 2014.

[90] Breiman, L. Random forests. *Machine Learning* Vol. 45, No. 1, 5–32, 2001.

[91] Gall, J.; Yao, A.; Razavi, N.; van Gool, L.; Lempitsky, V. Hough forests for object detection, tracking, and action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 33, No. 11, 2188–2202, 2011.

[92] Tejani, A.; Tang, D. H.; Kouskouridas, R.; Kim, T. K. Latent-class Hough forests for 3D object detection and pose estimation In: *Computer Vision – ECCV 2014. Lecture Notes in Computer Science, Vol. 8694*. Fleet, D.; Pajdla, T.; Schiele, B.; Tuytelaars, T. Eds. Springer Cham, 462–477, 2014.

[93] Coates, A.; Ng, A.; Lee, H. An analysis of single-layer networks in unsupervised feature learning. In: Proceedings of the 14th International Conference on Artificial Intelligence and Statistics, Vol. 15, 215–223, 2011.

[94] Kriegel, S.; Bodenmüller, T.; Suppa, M.; G. Hirzinger. A surface-based next-best-view approach for automated 3D model completion of unknown objects. In: Proceedings of the IEEE International Conference on Robotics and Automation, 4869–4874, 2011.

Rui Zeng received his B.Eng. degree from Hefei University of Technology, China, in 2017. He is currently pursuing a master degree in the Department of Computer Science and Technology, Tsinghua University, China. His research interests include computer vision, deep learning, and active robotic reconstruction.

Yuhui Wen received her Ph.D. degree from the University of the Chinese Academy of Sciences (UCAS), Beijing, China, in 2020. She is currently a postdoc at Tsinghua University. Her research interests include artificial intelligence, virtual reality, and human motion analysis.

Wang Zhao received his B.Eng. degree in electronic engineering from Tsinghua University in 2019. He is currently a Ph.D. student in the Department of Computer Science and Technology, Tsinghua University. His current research interests are computer vision, robotics, and machine learning.

Yong-Jin Liu is a professor in the Department of Computer Science and Technology, Tsinghua University. He received his B.Eng. degree from Tianjin University, China, in 1998, and his Ph.D. degree from Hong Kong University of Science and Technology in 2004. His research interests include computational geometry, computer vision, and computer graphics. He is a senior member of the IEEE and a member of ACM. For more information, visit http://cg.cs.tsinghua.edu.cn/people/Yongjin/Yongjin.htm.