Perception and Navigation in Autonomous Systems in the Era of Learning: A Survey

Yang Tang®, Senior Member, IEEE, Chaoqiang Zhao®, Jianrui Wang, Chongzhen Zhang®, Qiyu Sun®, Wei Xing Zheng®, Fellow, IEEE, Wenli Du®, Member, IEEE, Feng Qian®, Member, IEEE, and Jürgen Kurths®

Abstract— Autonomous systems possess the features of inferring their own state, understanding their surroundings, and performing autonomous navigation. With the applications of learning systems, like deep learning and reinforcement learning, the visual-based self-state estimation, environment perception, and navigation capabilities of autonomous systems have been efficiently addressed, and many new learning-based algorithms have surfaced with respect to autonomous visual perception and navigation. In this review, we focus on the applications of learning-based monocular approaches in ego-motion perception, environment perception, and navigation in autonomous systems, which is different from previous reviews that discussed traditional methods. First, we delineate the shortcomings of existing classical visual simultaneous localization and mapping (vSLAM) solutions, which demonstrate the necessity to integrate deep learning techniques. Second, we review the visual-based environmental perception and understanding methods based on deep learning, including deep learning-based monocular depth estimation, monocular ego-motion prediction, image enhancement, object detection, semantic segmentation, and their combinations with traditional vSLAM frameworks. Then, we focus on the visual navigation based on learning systems, mainly including reinforcement learning and deep reinforcement learning. Finally, we examine several challenges and promising directions discussed and concluded in related research of learning systems in the era of computer science and robotics.

Index Terms— Autonomous system, deep learning, environment perception, learning systems, navigation, reinforcement learning.

I. INTRODUCTION

In recent years, with the rapid developments in learning systems, such as deep learning and reinforcement learning, learning systems have been widely applied in various fields in smart grid [1], biology [2], finance [3], object detection [4], industrial production processes [5], and particularly in the autonomous systems of robots. Autonomous systems have gained a broad application prospect in various industries, including autonomous robots [6] and autonomous driving [7], [8]. Although current autonomous systems can perform single, simple, and repetitive tasks, such as aided driving [9] and transportation [10], the future of autonomous systems has significant potential. With the help of deep neural networks, autonomous systems that can learn and think like humans are becoming a reality. Intelligent and autonomous systems are the ultimate aim, which can perform advanced tasks autonomously, interact with humans, and even work better than humans [11]. Primarily, the autonomy of autonomous vehicle systems relies on the results of in-depth environment perception, intelligent motion planning, and accurate control [12]. The architecture of autonomous systems is illustrated in Fig. 1. Based on their perceiver [13], [14], autonomous systems understand their own state and surrounding environments by covering visual localization, mapping, and understanding the environment. Finally, autonomous systems can reach the designated position autonomously and complete advanced missions by combining the results of environment perception and motion planning with control signals.

Perceiving and understanding the environment are the basic elements of autonomous systems [13]. The development and application of visual simultaneous localization and mapping (vSLAM) have equipped robots with the ability to locate themselves and model the environment from vision, which has significantly expanded the autonomy and intelligence of robots. With the help of vSLAM, autonomous systems have the ability to use different visual sensors to collect environmental information, model their surroundings, and estimate their current state [15].
A. Perceiving the Environment

A good perception and understanding of the surrounding environment are indispensable for autonomous systems. vSLAM algorithms have been widely applied to model the environments into different types based on the actual requirements, including sparse map [16], semi-dense map [17], and dense map [18], as shown in Fig. 2(a)–(c). Although the geometric structures of surroundings in these representations are clearly perceived and modeled, a high-level information of these objects, like the semantic information, is still lacking.

B. Perceiving Their Own State

The state of an autonomous vehicle is described by its position and orientation. Understanding their current state in real-time is important for autonomous systems, which is the main precondition of autonomous control. Although current vSLAM algorithms play a crucial role in self-localization and ego-motion estimation, there are still some strong assumptions imposed in current vSLAM systems, such as the static scene hypothesis and the photometric consistency hypothesis.

C. Visual Navigation

The ability of autonomous navigation is also essential in autonomous systems. When an autonomous vehicle is assigned a destination, it requires the capabilities of planning a reasonable path or trajectory. Poor or untimely planning may lead to terrible results, such as collision and crash. Therefore, the ability of human-like planning is the future development direction, and it is possible to achieve this intention with the help of learning framework. Since traditional motion planning methods have been well summarized in [9], this review mainly focuses on the aspect of reinforcement learning-based navigation in autonomous systems.

D. Learning-Based Methods for Visual Perception and Navigation

With the development in learning framework [20], deep learning and reinforcement learning have demonstrated outstanding performance in image processing [21], [22], natural language processing [23], [24], motion estimation [25], game theory [26], biology [2], finance [3], and control [27], etc. The impact of learning framework on perception as well as navigation is transformational, and it has made significant advances in autonomous systems [13]. Recently, deep learning-based models are widely used in relevant works of environment perception, such as monocular depth estimation [28], ego-motion prediction [25], objective detection [4], and semantic segmentation [29]. Furthermore, to improve the tracking, localization, and mapping performance of current vSLAM methods in some complex environments (e.g., low light or nighttime scenes), attempts have been made to incorporate vSLAM with deep learning and satisfactory results have been obtained [30]. For example, some related works [31], [32] incorporated learning-based semantic understanding into the vSLAM to reconstruct the semantic maps of surroundings, as shown in Fig. 2(d), thereby getting a high-level understanding of surroundings. Moreover, related work in [33] has demonstrated that reinforcement learning exhibits good performance in robotic navigation. It resolved and implemented the navigation problems in an end-to-end manner. In addition, reinforcement learning enables robots to learn and imitate humans to make decisions. Unlike some well-written reviews [13], [15], [34], this survey mainly focuses on surveying the learning-based perception, including self-state perception and environment perception, as well as the representative results for reinforcement learning-based navigation in autonomous systems.

The rest of the article is organized as follows: Section II introduces related works on visual perception, including a brief review of traditional vSLAM methods, deep learning-based visual perception, and methods combining deep learning with vSLAM. Section III provides an overview of the reinforcement learning-based visual navigation. Section IV summarizes the deficiencies and challenges of existing learning systems for visual perception and navigation, and provides some ideas about future directions. Finally, this survey is concluded in Section V.
be efficiently solved by vSLAM algorithms or sub-topics of vSLAM algorithms. Some classic simultaneous localization and mapping (SLAM) methods are well summarized and discussed in [13] and [34]. Cadena et al. [13] reviewed the related works on SLAM over the last 30 years in detail. They revisited and answered several important and meaningful questions related to SLAM and stated that “SLAM is necessary for autonomous robots.” Different from previous review articles, in this section, we mainly focus on the application of deep learning algorithms in perception by subdividing them into three types.

### A. Geometric Methods-Based Visual Perception

SLAM is a common perception method in current autonomous systems. Compared with the SLAM systems that use Lidar sensors [74], [75], visual sensors such as RGB cameras [49], [76] can provide more environmental information, and they have been extensively investigated in recent years owing to their portability. Therefore, we briefly summarize different types of vSLAM methods in a chronological order first, as presented in Table I. Their categories of optimization, maps, and sensors are enumerated in detail. From Table I, we find that filtering-based vSLAM methods have been widely studied in the initial stage owing to their low computational burden. With the development in computer science, optimization-based vSLAM methods have become popular in recent years due to their higher accuracy. Meanwhile, dense maps are usually constructed by direct methods based on RGB-D sensors, like [44], [45], and [77]. In addition, new sensors, such as event cameras, and multisensor data fusion are attracting significant attention and research prospects [50], [54], [78], [79]. In this section, we communicate the basic principles of the three classical monocular vSLAM solutions, including feature-based methods [52], direct methods [17], and semi-direct methods [49]. The main difference between these three methods is the pose optimization by minimizing either the reprojection error, photometric error, or both [76].

Feature-based methods have dominated vSLAM for a long time, and different man-made features (like SIFT [80], SURF [81], and ORB [82]) have been designed to improve their robust tracking and mapping in different scenarios. The feature-based methods can be divided into three parts, including image input, feature extraction and matching, and tracking and mapping. Most recently, ORB-SLAM3 [73] is proposed to support different kinds of sensors, like monocular, stereo, RGB-D, and IMU sensors, and it also supports a variety of camera models. ORB-SLAM3 system is much more versatile, accurate, and robust than previous work. However, the performance of feature-based methods relies on the correct matching, and they will fail to initialize and track in low-texture and repeated-texture scenes [17] because of mismatch, suffers from the divergence in the optimization algorithm, and accumulation of drift. Direct methods cancel the process of feature extraction and matching, and the photometric information of pixels is directly used for pose and depth calculations during tracking and mapping [17], [83]. Direct methods regard the pose estimation as a nonlinear optimization problem and iteratively optimize the initial motion guess by minimizing the photometric error [17]. Therefore, direct methods rely heavily

---

**Table I**  
**SUMMARY OF MAJOR GEOMETRIC V SLAM METHODS. “MONO” DENOTES THE MONOCULAR CAMERA, AND “STEREO” STANDS FOR STEREO CAMERA**

| Year | Reference | Method | Type | Map | Sensor |
|------|-----------|--------|------|-----|--------|
| 2003 | Real-time SLAM [35] | ✔ | ✔ | ✔ | ✔ | Mono. |
| 2004 | Davison et al. [36] | ✔ | ✔ | ✔ | ✔ | Mono. |
| 2005 | CV-SLAM [37] | ✔ | ✔ | ✔ | ✔ | Mono. |
| 2006 | Smith et al. [38] | ✔ | ✔ | ✔ | ✔ | Mono. |
| 2007 | Mono-SLAM [39] | ✔ | ✔ | ✔ | ✔ | Mono. |
| 2007 | PTAM [40] | ✔ | ✔ | ✔ | ✔ | Mono. |
| 2008 | Silvestre et al. [41] | ✔ | ✔ | ✔ | ✔ | Mono. |
| 2009 | Migliore et al. [42] | ✔ | ✔ | ✔ | ✔ | Mono. |
| 2010 | Newman et al. [43] | ✔ | ✔ | ✔ | ✔ | Mono. |
| 2011 | DTAM [44] | ✔ | ✔ | ✔ | ✔ | RGB-D |
| 2011 | KinectFusion [45] | ✔ | ✔ | ✔ | ✔ | Mono. |
| 2012 | Kin remix [46] | ✔ | ✔ | ✔ | ✔ | RGB-D |
| 2013 | Weiskircher et al. [47] | ✔ | ✔ | ✔ | ✔ | Mono. |
| 2013 | Ender et al. [48] | ✔ | ✔ | ✔ | ✔ | Mono. |
| 2014 | Li et al. [49] | ✔ | ✔ | ✔ | ✔ | Mono. |
| 2014 | SVO [50] | ✔ | ✔ | ✔ | ✔ | Mono. |
| 2014 | LSD-SLAM [51] | ✔ | ✔ | ✔ | ✔ | Stereo |
| 2014 | Weiskircher et al. [52] | ✔ | ✔ | ✔ | ✔ | RGB-D, Event camera |
| 2015 | Stereo-LSD-SLAM [53] | ✔ | ✔ | ✔ | ✔ | Stereo |
| 2015 | ORB-SLAM [54] | ✔ | ✔ | ✔ | ✔ | Stereo |
| 2015 | Leutenegger et al. [55] | ✔ | ✔ | ✔ | ✔ | Mono. |
| 2015 | Bobick et al. [56] | ✔ | ✔ | ✔ | ✔ | Mono. |
| 2016 | Essential Fusion [57] | ✔ | ✔ | ✔ | ✔ | RGB-D |
| 2016 | Finger et al. [58] | ✔ | ✔ | ✔ | ✔ | Mono. |
| 2016 | SVO 2.0 [59] | ✔ | ✔ | ✔ | ✔ | Mono. |
| 2016 | EVO [60] | ✔ | ✔ | ✔ | ✔ | Mono. |
| 2017 | DSO series [61]-[63] | ✔ | ✔ | ✔ | ✔ | Stereo, RGB-D |
| 2017 | ORB-SLAM2 [64] | ✔ | ✔ | ✔ | ✔ | Mono. |
| 2017 | BundleFusion [65] | ✔ | ✔ | ✔ | ✔ | RGB-D |
| 2017 | Mor et al. [66] | ✔ | ✔ | ✔ | ✔ | RGB-D |
| 2018 | ProSLAM [67] | ✔ | ✔ | ✔ | ✔ | Mono. |
| 2018 | Sun et al. [68] | ✔ | ✔ | ✔ | ✔ | Stereo |
| 2018 | iCEPA [69] | ✔ | ✔ | ✔ | ✔ | Mono. |
| 2018 | VINS-mono [70] | ✔ | ✔ | ✔ | ✔ | Stereo |
| 2018 | Lee et al. [71] | ✔ | ✔ | ✔ | ✔ | Stereo |
| 2019 | BAD-SLAM [72] | ✔ | ✔ | ✔ | ✔ | Stereo |
| 2019 | RESLAM [73] | ✔ | ✔ | ✔ | ✔ | Stereo |
| 2020 | Hoang et al. [74] | ✔ | ✔ | ✔ | ✔ | Stereo |
| 2021 | OVS-SLAM [75] | ✔ | ✔ | ✔ | ✔ | Stereo |
| 2021 | ORB-SLAM3 [76] | ✔ | ✔ | ✔ | ✔ | Stereo, RGB-D, IMU |

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
on the luminosity consistency assumption [84]. Semi-direct methods first establish feature correspondences on the basis of direct methods, which is the main difference from other methods [49], [57]. The principle of epipolar line constraint is applied to match the same features on the epipolar line. After matching the features, the solved pose is optimized by minimizing the reprojection error. Therefore, semi-direct methods handle the tracking problem by minimizing the photometric error and the reprojection error. Similar to direct methods, semi-direct methods have a high requirement on image quality and are sensitive to photometric changes.

Although the architecture of vSLAM algorithms has been very maturely over the past 30 years and the three kinds of the above-mentioned approaches have achieved good performance in normal indoor/outdoor scenes, their tracking robustness and localization/re-localization accuracy in many complex scenarios (like high-dynamic/large scale/nighttime environments or across weather/across-seasons conditions) still need to be further improved [13], [84], [85].

In conclusion, although traditional geometry-based vSLAM methods have achieved amazing results in environmental mapping and self-localization, these methods still have some shortcomings. For example, feature-based methods cannot adapt to low texture area; direct methods need a good initialization; semi-direct methods are sensitive to luminosity; traditional vSLAM/VO methods cannot handle changing lighting/weather/season conditions; monocular vSLAM/VO methods suffer from scale ambiguity, and so on [84], [86], [87]. With the continuous development of deep learning in image processing, applying the latest deep learning systems to the existing vSLAM to handle the current problems in vSLAM methods is evolving into a popular research field.

B. Deep Learning-Based Visual Perception

With the development in deep learning, utilizing deep neural networks to address computer vision tasks has evolved into a popular research field in recent years. Many sub-topics of vSLAM for environment perception have been extensively studied based on deep learning, such as monocular depth and ego-motion prediction, which will be specified in the following.

1) Learning-Based Monocular Depth Perception: Depth is one of the most important information for autonomous systems in scene reconstruction, self-localization, obstacle avoidance, and so on. Although the active depth sensors are available for depth perception, image-based techniques are often preferred thanks to the increasing availability of standard cameras on most consumer devices [144]. Structure-from-motion (SfM) [145], [146] and stereo matching [147], [148] are two of the most popular methods to recover the depth from sequential or left and right images [147], and the depth is calculated by the triangulation and continuously optimized by projection cost and matching cost. However, the above methods rely on the assumption that multiple observation of the scene are available [99], which means that the above methods are not well applicable to estimating depth from a single image.

Estimating the depth from a single image is an ill-posed problem [88], which requires significant man-made prior knowledge when handled by traditional geometric methods [149], [150]. Deep neural networks can recover pixel-level depth information from single images in an end-to-end manner based on the prior knowledge learned from ground truth depth labels or geometric relationships between images [151]. Since both ground-truth-based supervised methods and geometry-based unsupervised methods have been well summarized in [151], in this article, we will focus on the latest work, starting with issues that remain unresolved in monocular depth estimation. As shown in [151], monocular depth estimation has made great progress in recent few years, and unsupervised methods are already close to that of supervised methods. After several years of development, the framework of monocular depth estimation has become very mature. Recent work focuses on improving the deficiencies of the existing unsupervised framework, like static scenario assumptions [152] and photometric consistency assumptions [129].

The smoothness loss is one of the most widely used constraints [28], [122], [123] to promote the smoothness of the surface depth of the object, for example, the depth of the adjacent points on the road surface varies by gradient. However, the existing methods do not impose smoothness constraints after distinguishing different targets in the scenario, resulting in smoothing edge areas that should be sharp in the estimated depth map. To address this problem, Yin et al. [153] proposed a novel geometric constraint to improve the accuracy of depth estimation as well as the geometric shape in the predicted depth map by considering the surface normal. Instead of using additional constraints to get a clear geometric structure in monocular depth estimation, the method proposed in [96] predicted a 2-D displacement field of the given depth map to re-sample pixels around the occlusion boundaries into sharper reconstructions. A recent study [131] showed that incorporating sequence information into monocular framework is helpful to improve depth prediction when the sequence information is available. Instead of estimating the accurate depth of each pixel, predicting the relative depth of pixels in the image is also crucial for scene perception and understanding [154], which can also obtain good results in recovering metric depth. When considering the widely application scenarios of high-resolution depth maps, like object detection and semantic segmentation, Miangoleh et al. [155] proposed to infer high-resolution depth maps from images based on pre-trained depth models.

Since the supervised signal of unsupervised methods is mainly based on the view reconstruction loss [28], view reconstruction relies heavily on static scenario assumptions. Therefore, these methods fail to predict depth for moving objects. To deal with this challenge, Godard et al. [156] designed an “Auto-Masking” to selectively eliminate pixels that keep the same position with same RGB value between adjacent frames in the sequence. However, this method can only eliminate the influence of objects moving at equivalent relative translation to the camera, while other dynamic objects will still have negative influence on the unsupervised training process. Therefore, with the help of semantic segmentation, Klingner et al. [157] divided the dynamic and static objects...
by the correspondence of class labels between frames, which is calculated by projection. Then, they eliminated the effects of these dynamic regions on view reconstruction loss.

There are also some novel studies that improve the accuracy of monocular depth estimation by utilizing the novel network framework, such as proposing novel depth network [158] or using novel attention mechanism [95], [159]. Introducing traditional geometry is also a good way, Wang et al. [110] tried to get a better pose estimation by using direct methods during training. The direct method was used to further optimize the output of the pose network before training, thereby getting a more accurate pose and depth estimation. Depth estimation based on novel cameras, like event-based camera [160], fish-eye camera [152], and panorama camera [161], is attracting more attention because of its advantages, like low latency and wide field-of-view. Inspired by the high performance of HRNet [162], Zhou et al. [133] introduced the HRNet into the unsupervised monocular depth estimation task and obtained satisfactory results.

Monocular depth estimation in special scenarios, such as adverse weather conditions and nighttime scenes, is gradually being focused. Because of the complex luminosity changes and photometric inconsistency at night, the previous unsupervised frameworks driven by view reconstruction consistency [28], [122], [123] cannot be applied to the nighttime scene [129] directly. Recent studies have tried to address this problem by using warped feature consistency [129] or cross-domain feature adaptation [127], which achieved good accuracy in nighttime depth estimation. Spencer et al. [129] designed a DeFeat-Net to simultaneously learn the cross-domain dense feature representations of frames. Moreover, a robust feature reconstruction consistency instead of view reconstruction consistency is used as the main supervised signal for the training of framework, thereby being able to adapt special scenarios. Based on the auto-encoder depth network pre-trained on day time, Vankadari et al. [127] used an additional nighttime encoder to encode the images of nighttime. A PatchGAN-based adversarial discriminator was designed to constrain the consistency between the features among the images of day time and nighttime, which are encoded by two encoders, respectively. Hence, the pretrained decoder can directly recover a depth map of a night-time image from features encoded by the night-time encoder. Zhao et al. [163] proposed to use a cyclegan-based domain adaptation framework to get an end-to-end night-time depth model from a pretrained day-time model, and it got a better results in night and even rainy night. Instead of using adaptation methods, Wang et al. [164] leveraged a mapping-consistent image enhancement module to deal with the low visibility and a statistics-based mask (SBM) to tackle textureless regions, so their work can directly train the model on night-time image sequences.

2) **Learning-Based Monocular Ego-Motion Perception**

Visual odometry (VO) is the process of estimating the ego-motion of an agent (e.g., vehicle, human, and robot) by using the input of a single or multiple attached cameras [34]. Geometry-based monocular VO methods handle the localization and tracking by minimizing the photometric error [59] or reprojection error [52] on sequential images. The difference between traditional VO and vSLAM is that VO system lacks the loop-closure detection and global optimization [34]. With the development of deep learning systems, using the features extracted by deep neural networks to regress the ego-motion in an end-to-end manner is becoming a hot application in recent years [25]. Compared with traditional VO methods, pose networks do not require complex parameter tuning, such as the settings of key frames and features [25]. Moreover, pose networks can learn the scale information from the ground truth during training, so these methods solve the monocular scale ambiguity problem that widely existed in traditional monocular VO methods [52], [59]. Konda and Memisevic [134] first estimated the motion information through deep learning-based methods by formulating pose prediction as a classification problem. Kendall et al. [25] first demonstrated the ability of convolutional neural networks (CNNs) on six-DOF pose regression. A deep CNN framework called PoseNet was designed for regressing monocular camera pose that could operate in different scenes in real-time. In [135], Costante et al. also used a deep CNN to learn high-level feature representation, and the major difference from [25] is that the dense optical flow was calculated and used to estimate the ego-motion instead of feeding RGB images into the CNN directly. Considering the dynamics and relations between adjacent pose transformations, Wang et al. [136] and Xue et al. [137] used recurrent neural networks (RNNs) for camera localization. Then, Xue et al. [138] further extended their work by incorporating two helpful modules named “Memory” and “Refining” into VO tasks, which outperformed the previous deep learning-based VO methods [137].

As the learning system is constantly evolving, introducing new learning architecture to current tasks has been a good way to improve the ability of pose network in high-level feature extraction and pose regression. Xue et al. [141] proposed to construct a view graph to excavate the information of the whole given sequence for absolute camera pose estimation, and a graph neural network was applied to model the total graph. Li et al. [126] introduced online meta-learning algorithms into previous learning framework, so that their method can continuously adapt to unseen environments in a self-supervised manner. Considering the error accumulation problem commonly suffered by previous learning-based methods, Zou et al. [165] tried to aggregate long-term temporal information by using convolutional long short term memory (Conv-LSTM) to model long-term temporal dependency. Meanwhile, long-range constraints based on long-range image snippets are used to improve temporal consistency over long sequences, just like the local optimization (bundle adjustment) that widely used in traditional VO methods. Chi et al. [115] studied the performance difference between feature-level collaboration and loss-level joint optimization for multitask learning (depth, pose, and optical flow), and feature-level collaboration shows much greater performance improvement for all three tasks. Therefore, they designed a single network to integrate all the three tasks, and the pose component regresses pose from both images and estimated disparity map and optical flow. Inspired by bundle adjustment, Wei et al. [142] proposed
a deep learning framework that iteratively improves both depth and pose based on the cost volume explicitly built to measure photo-consistency and geometric-consistency. Zhuang and Chandraker [143] presented an uncertainty-based probabilistic framework that integrating pose predictions from deep neural networks and solutions from geometric feature-based solvers (five-point method and bundle adjustment). Instead of estimating poses from images, Zhao et al. [166] recovered relative pose by directly solving the fundamental matrix from dense optical flow correspondence, which was predicted by an optical flow network, and the results demonstrated the effectiveness of the framework in pose estimation. Jiao et al. [116] obtained the pose between frames by minimizing the reprojection error since the optical flow and depth are predicted by deep neural networks.

The traditional methods have proved that combining visual information with inertial information is helpful for improving the visual localization accuracy [53], [167], [168]. However, these visual-inertial odometry (VIO) methods suffer from accurate calibration between sensors, time-stamp synchronization between inertial and visual data, and effective inertial and visual information fusion [53], [167], [168]. Researchers believe that inertial information is also helpful in learning-based methods. Therefore, Clark et al. [139] proposed the first end-to-end VIO framework based on deep learning without the need for time-stamp alignment and manual calibration between different sensors. They used the CNN architecture to extract visual features and long short-term memory (LSTM) to extract the inertial features, and fused their features using a core LSTM processing module for pose regression. For a better integration of visual and inertial features extracted by the deep neural networks, Chen et al. [140] presented a selective sensor fusion framework based on the attention mechanism, which autonomously selects the most useful features extracted from images or inertial measurement unit (IMU) by deep neural network. Therefore, even in the case of poor image quality, their algorithm can get accurate poses with the help of inertial data.

We briefly summarize the deep learning-based monocular depth and ego-motion estimation according to their published years, the training data, the training mode, and the missions, as shown in Table II. From the table, we find that attention has been paid increasingly to unsupervised methods these years because unsupervised methods do not require expensive ground truth [28]. Besides, considering the in-depth relationship between projection and optical flow between frames, researchers always extend the unsupervised pose and depth estimation framework with optical flow estimation [121], [122]. Recently, scene flow (optical flow in 3-D space) estimation [114], [116] is getting more attention, which is trained together with depth and pose network in an unsupervised manner. Since optical flow, scene flow, depth, and pose are tightly coupled, the training strategy will have an impact on the performance of each task [116]. Therefore, the multitask frameworks have become popular in recent years, and the geometric relationship between these tasks (flow, segmentation, mask) has been exploited to improve the performance of the network. Besides, the data from multisensors (like camera and IMU) have also been added to join the network to provide additional information [109], [140], thus promoting the training of networks.

C. Deep Learning With vSLAM

The methods combining vSLAM with deep learning have also been extensively studied and have led to notable improvements to traditional vSLAM methods, like tackling the scale ambiguity of monocular vSLAM [169], [170], improving the robust tracking and accurate mapping of vSLAM [171], [172], and extending the semantic perception of the environments [174], [175].

1) Learning-Based Monocular Depth Estimation and vSLAM: Depth information plays an important role in traditional vSLAM methods, and sensor-based and triangulation-based methods are two basic ways to obtain the depth of features. With the development of deep learning in the field of monocular depth estimation, researchers are trying to use deep learning-based methods as an alternative to the traditional depth calculation methods of vSLAM. The combination of deep learning-based depth estimation and traditional vSLAM methods has been proven to be effective in obtaining the depth of features and overcoming the monocular scale ambiguity, thereby improving mapping and replacing the RGB-D sensors [169], [176]. Depth prediction was first introduced to dense monocular vSLAM by Laina et al. [176]. Since the mapping process reduces the dependence on feature extraction and matching, this method has the potential to reconstruct low-texture scenes. Moreover, this work showed that the depth estimation network can replace the depth sensors (such as RGB-D) and can be used for dense reconstruction. After that, a real-time dense vSLAM framework was proposed in [169]. They used the LSD-SLAM [17] as the baseline and fused the depth estimation and semantic information. Unlike the work by Laina et al. [176], where the depth estimation was directly used in vSLAM, Tateno et al. [169] considered the predicted depth map as the initial guess of LSD-SLAM, and further refined the predicted depth value by the local or global optimization algorithms in vSLAM. This method not only got a higher pose accuracy than LSD-SLAM, but also overcame the issue of scale inconsistency in dense monocular reconstruction. Similarly, Yang et al. [171] proposed a novel semi-supervised disparity estimation network and incorporated it into direct sparse odometry (DSO) [88], thereby achieving a better accuracy to monocular DSO and attaining a comparable performance to previous stereo DSO methods. Recently, Loo et al. [177] presented a CNN-SVO pipeline that leveraging the SVO [49] with depth prediction network to improve the mapping and tracking of SVO. Czarnowski et al. [178] proposed a real-time probabilistic dense vSLAM system that integrates learned priors (depth) over geometry with classical vSLAM formulations in a probabilistic factor-graph formulation, and got a better accuracy than [169] in both trajectory and depth estimation. Combining depth estimation with vSLAM has been proven to effectively improve the performance of traditional monocular vSLAM. Moreover, vSLAM can also be used to promote the accuracy of depth networks. For
example, Tawari et al. [179] proposed a self-improving framework. On the one hand, the predicted depth was used to perform RGB-D feature-based vSLAM. On the other hand, the pose calculated by RGB-D feature-based vSLAM instead of that predicted by pose network was leveraged to train the depth network, thereby leading to more accurate depth estimation. The above works have shown how to integrate vSLAM with depth prediction via a deep neural network, and it is a promising direction to address inherent limitations of traditional vSLAM, especially with respect to estimating the absolute scale and obtaining dense depths.

2) Learning-Based Pose Estimation and vSLAM: Although pose networks have achieved real-time performance and satisfactory accuracy, the existing learning-based pose estimation methods do not include the mapping thread [25], [136], which is important for the perception of the environmental structure. Besides, traditional direct methods rely heavily on the initial guess of pose during tracking, resulting in instable initialization and inaccurate tracking [84], [88]. Therefore, combining learning-based pose estimation with traditional vSLAM is a good way to overcome the above deficiencies [172], [180]. Zhao et al. [172] designed a self-supervised pose prediction network and incorporated it into DSO [88]. They considered the output of the pose network as the initial pose guess of direct VO, which replaced the constant motion model used in DSO; then, the initial pose was improved by the nonlinear optimization in DSO. This method got a more robust initialization and tracking than traditional DSO when testing on the KITTI odometry sequences [181]. Yang et al. [180] also focused on this field, and they proposed a novel framework for monocular VO that exploits deep networks on three levels—deep depth, pose, and uncertainty estimation, which not only improve the robust initialization and tracking of DSO in the challenging scenarios with photometric changes but also assist in recovering the scale information of monocular VO. Different from the above frameworks, Wagstaff et al. [182] proposed to use a deep neural network to correct the pose estimated by traditional VO frameworks, and a self-supervised deep pose
3) Learning-Based Image Enhancement and vSLAM: Current monocular vSLAM methods have achieved good robustness under specific scenarios, such as outdoor sunny scenes with normal illumination conditions [181]. While driving in complex environments, such as during the night, in rain, and other scenarios, current monocular vSLAM systems cannot accurately estimate the pose of robots and reconstruct the point clouds of environments. For example, compared with the day-time scenarios under a single light source (sun), the night-time scenarios suffer from complex lighting changes because of multiple light sources (e.g., street lights, own car light, and other car lights) [127], which affects the extraction of high-quality points and the accuracy of feature matching between frames [30]. Therefore, tracking the key points or features between images in such scenarios is unstable and in accuracy because of the changing illumination, resulting in inaccurate calculation of depth and pose [30]. Besides, different weather conditions in the same scenario also cause changes in the luminosity and feature descriptors of the scene, which brings challenges to feature matching and relocalization [85]. Researchers have proposed some geometric methods to improve the performance of vSLAM in challenging environments, such as using multiple cameras [184], designing a new NID metric [85], and raising novel feature descriptor [185], and these methods can achieve robust VO and relocalization under different lighting and weather conditions.

With the development of learning systems in image style translation [186], [187] and video synthesis [188], [189], deep learning-based image enhancement provides a new and simple way for vSLAM systems to overcome challenging environments [190], [191]. Deep learning-based image enhancement helps to enhance the quality of images, like enhancing the brightness constancy of images [192] or transferring the images from low light to normal light [30], so as to make the images more suitable for current vSLAM systems. Considering that direct methods cannot handle the dynamic lighting changes, Gomez-Ojeda et al. [192] used deep neural networks to enhance the brightness constancy of image sequences captured from high dynamic range (HDR) environments. The experiments showed that learning-based image enhancement can improve the trajectory estimation of ORB-SLAM [52] and DSO [88] in HDR environments. Since the illumination influences feature extraction and matching, Jung et al. [30] proposed a new framework called multiframe GAN that translated the images from night-time to day-time to improve the quality of input images. Both stereo ORB-SLAM [16] and stereo DSO [60] achieved accurate tracking performance on the transferred high-quality day-time images, which means that their method [30] can overcome the low light environments. Unlike Jung et al. [30], von Stumberg et al. [84] replaced the input of direct methods (gray-scale images) with feature maps created by their designed GN-Net for relocalization tracking.

Since GN-Net can predict the consistent feature map of the same scene under different lighting and weather conditions, their method had the ability to achieve accurate tracking and relocalization in different weather conditions.

4) Learning-Based Object Detection, Semantic Segmentation, and vSLAM: We consider the following three problems.

a) Dynamic scene adaptability: Traditional vSLAM relies heavily on static scene assumption, i.e., the performance of vSLAM is limited by moving objects [193], as shown in Fig. 3(a). Both photometric error and reprojection error are based on geometric projections between frames, but the features on the dynamic object do not satisfy the projection relationship based on the camera motion, which will lead to inaccurate pose estimation. The features on static objects are positive to improve the accuracy, while those on dynamic objects have a negative impact on the tracking process [195]. Therefore, if the dynamic objects on the input images can be detected and labeled, this problem will be well addressed. Considering the outstanding performance of deep learning-based object detection and semantic segmentation, the integration of deep learning framework and vSLAM can effectively assist vSLAM in identifying dynamic objects in the environment to classify and handle the dynamic features. Excellent detection and segmentation networks, such as YOLO [196], SSD [197], Mask-RCNN, [198], and SegNet [199], have been incorporated into traditional vSLAM frameworks as an additional thread to identify and eliminate the dynamic features. Zhong et al. [200] presented a novel system that integrated vSLAM with the object detector SSD, called Detect-SLAM. The SSD was used to detect the dynamic and static objects for every key frame; since the extracted features on the dynamic objects were removed, the remaining static features satisfied the projection function between frames, which greatly improved the accuracy of the pose and depth solutions. Wang et al. [173] considered the effects of moving objects on localization accuracy and constructed maps, and developed a novel vSLAM solution. They used YOLOv3 [201] to detect moving objects and constructed a semantic static map with the data without moving objects. Xiao et al. [195] developed a new detection thread to detect and remove the dynamic objects, and designed a selective tracking algorithm to process the dynamic features during tracking. Because the object detection methods are not considered during pixel-level semantic annotation, the classification of feature attributes is
not accurate enough. Therefore, Yu et al. [202] presented a robust semantic vSLAM for dynamic environments with five threads based on ORB-SLAM2. They used SegNet to segment the movable objects at the pixel level and designed a moving consistency check process to detect the movements of the movable ORB features. Only the semantically and geometrically dynamic features were deleted. Similarly, Cui and Ma [193] combined the results of semantic segmentation from SegNet with ORB-SLAM2. They proposed a new method, called semantic optical flow (SOF), to improve the detection of dynamic features and reasonably remove the dynamic features during tracking. Unlike the works [173], [193], [195], [200], [202] that directly detect and delete the dynamic features, recent studies tried to further estimate and utilize the dynamic objects in the scenes [203], [204]. Huang et al. [203] proposed a stereo VO framework that not only estimated the motion of camera but also clustered the surrounding objects. A sliding window optimization was used to solve the motions of camera and surrounding dynamic objects. Yang and Scherer [204] dug deeper into the relationship between the motion of camera and surrounding objects, and found that the two parts can improve each other. Since both dynamic and static objects can provide long-range geometric and scale constraint, it is helpful to improve the motion of camera but also clustered the surrounding objects. A sliding window optimization was used to solve the motions of camera and surrounding dynamic objects. Yang and Scherer [204] dug deeper into the relationship between the motion of camera and surrounding objects, and found that the two parts can improve each other. Since both dynamic and static objects can provide long-range geometric and scale constraint, it is helpful to improve the camera pose estimation and constrain the monocular drift.

b) Scale recovery and visual localization: Scale ambiguity has always been a big challenge for monocular vSLAM, which brings great uncertainty to accurate trajectory prediction and mapping [87]. Because objects in reality have their own inherent properties, like the height of cars, these properties can be used for monocular vSLAM to get the absolute scale information of scenes. Therefore, semantic information can be utilized to build a bridge between objects and their properties, and it has shown its effectiveness in monocular vSLAM for scale recovery and assisting localization. Semantic information introduces the size information of objects in the environment into the vSLAM framework to handle the problem of monocular scale ambiguity. Frost et al. [205] represented objects in the environment as spheres and recovered the scale from the detected objects with a known radius. Similarly, Sucar and Hayet [170] recovered the scale by setting the prior height of the object (car). A detection method was used to detect this object and compute the height, and the scale was solved by the ratio of the calculated height to the prior height. For localization, Stenborg et al. [206] proposed a novel method that localizes the camera based on semantically segmented images, which is different from traditional localization methods based on features. To obtain more accurate localization, Bowman et al. [207] first integrated the geometric, semantic, and IMU information into a single optimization framework and then associated scale information with semantic information. Lianos et al. [208] utilized the semantic information of the scenes to establish mid-term constraints in the tracking process, thereby reducing the monocular drift in VO.

High-level semantic perception: Autonomous systems need to be able to perform high-level tasks, while the point cloud maps built by traditional vSLAM cannot fully meet the requirements of these tasks. Therefore, a multilevel understanding of their surroundings is essential. For instance, autonomous vehicles should have an understanding of the areas that are drivable and those that have obstacles. However, the environments modeled by traditional vSLAM methods are represented by point clouds, which only contain the location of the point and cannot provide any high-level information about 3-D objects. Although the current metric representation for vSLAM executes some basic tasks, such as localization and path planning, it is still insufficient for some advanced tasks, such as human–robot interaction, 3-D object detection, and tracking. Therefore, high-level and expressive representations will play a key role in the perception of autonomous systems. To obtain high-level perception, an object-level environment representation [209] was proposed in 2011 by modeling the objects in advance and matching them in a global point cloud map. Salas-Moreno et al. [210] extended this work in [209]. They created an object database to store the 3-D models generated by Kinectfusion [44] and computed the global descriptor of every object model for quick matching based on [211]. They also demonstrated that object-level mapping is useful for accurate relocalization and loop detection. Contrary to building the models in advance, Sunderhauf et al. [212] proposed an online modeling method for generating the point cloud models of objects, along with a novel framework for vSLAM by combining object detection with data association to obtain semantic maps. However, traditional geometry-based high-level environment perception requires modeling and matching objects in the environment in advance, which leads to the complexity of the whole process, i.e., only some objects can be modeled and recognized in these methods.

In comparison to an object-level maps, pixel-level semantic maps based on learning systems are more precise because they present the semantic information of each point in the maps. To improve the accuracy of segmentation and semantic mapping, conditional random fields (CRFs) have been widely used in related works. A voxel-CRF model was presented in [213] to associate the semantic information with 3-D geometric structure, and a dense voxel-based map with semantic labels was constructed. For consistent 3-D semantic reconstruction, Hermans et al. [214] proposed a novel 2-D–3-D label transfer method based on CRFs and Bayesian updates. Considering the intrinsic relationship between geometry and semantics, Kundu et al. [215] utilized the constraints and jointly optimized semantic segmentation with 3-D reconstruction based on CRFs. Gan et al. [32] focused on the continuity of maps and valid queries at different resolutions, and exploited the sparse Bayesian inference for accurate multiclass classification and dense probabilistic semantic mapping. With the help of semantic maps, autonomous systems can obtain a high-level understanding of their surroundings, and they can easily know “which and where is the desk.”

With the development in deep neural networks, several detection and segmentation methods based on deep learning are proposed. Methods for object detection and image segmentation have been reviewed in [4] and [29]. Leveraging deep learning-based image segmentation to perform semantic mapping is also a hot topic. Li et al. [216] combined the LSD-SLAM [17] with CNN-based image segmentation to
reconstruct a semi-dense semantic map. Cheng et al. [174] integrated a CRF-RNN-based segmentation algorithm with ORB-SLAM [52], and built a dense semantic point-cloud map by using RGB-D data. Deep learning-based semantic segmentation with dense SLAM frameworks have also been applied to construct dense semantic maps. McCormac et al. [175] incorporated CNN-based semantic prediction into state-of-the-art dense vSLAM method, ElasticFusion [55]. They considered the multiview segmentation result of the same 3-D point and fuse semantic information in a probabilistic manner.

When/where can we integrate learning methods to aid traditional frameworks, like vSLAM? There are two main ways of using deep learning to improve traditional frameworks: one is to enhance the quality of inputs through the learning systems, like image enhancement and vSLAM; the other is to embed the learning systems into traditional frameworks, like pose estimation, depth estimation and vSLAM. For example, considering that traditional vSLAM cannot well adapt to challenging low-light environments, learning methods are used to enhance the stability of feature tracking of vSLAM by improving the quality of input images [30]. Since dynamic objects will affect the feature matching, which in turn affects the pose and depth solution of vSLAM, learning systems are used to detect dynamic objects and help to eliminate the dynamic features [195]. Therefore, the basic idea is to analyze the limitations and shortcomings of traditional vSLAM, and then introduce learning systems to improve the traditional vSLAM framework. In addition, we should also note that the introduction of the learning systems also brings some problems to the entire framework, such as the increase of computation, the dataset dependence of the learning systems, etc., and there also remain problems we need to address in the future.

III. AUTONOMOUS VISUAL NAVIGATION

After perceiving the surroundings and state, autonomous robots will plan appropriate trajectories according to the missions, their own state as well as the environmental information. A survey of geometry-based motion and control planning for autonomous vehicles is proposed in [9]. Therefore, in this section, we mainly focus on autonomous visual navigation based on reinforcement learning, as shown in Fig. 4. We first present visual navigation methods and introduce three main deep reinforcement learning methods. Then, we review deep reinforcement learning-based visual navigation scenarios, methods, and environments.

Navigation can be defined as a process of accurately determining one’s location, planning, and following a route from one place to another. With the help of the advanced sensors and navigation algorithms, vision has been introduced into navigation [217], [218]. Compared with other navigation methods, such as magnetic navigation [219], inertial navigation [220], laser navigation [221], and GPS navigation [222], visual navigation has a relatively low cost and general simulation platforms. Therefore, visual navigation has become a mainstream research approach for researchers. Traditional visual navigation of mobile robots is generally based on three main methods: map-based navigation, map-building-based navigation, and mapless navigation [223].

Map-based navigation requires the global map of the current environment to make decisions for navigation. For example, in [224], the robot used a generic map to accomplish symbolic navigation. Specifically, the robot was not guided to the locations with specific coordinates but with symbolic commands. Symbolic commands are the general description of the types of entities in the environment. In map-building-based navigation, robots use different sensors to perceive the environment and update the map. For example, in [225], the robot accomplished long-distance navigation with the help of a topological map. Specifically, the global environment was built as a topological map and described by graphics during navigation. An appearance-based system and a visual servoing strategy qualitatively estimated the position of the robot and kept it on a specific trajectory employing omnidirectional cameras. In mapless navigation, robots do not have any environment information and navigate with the perceived information without maps. Saeedi et al. [226] presented a general-purpose 3-D trajectory-tracking system. This system could be applied to unknown indoor and outdoor environments without the need of mapping the scene, odometry, or the sensors other than vision sensors.

A. Reinforcement Learning-Based Visual Navigation

Since reinforcement learning is suitable for continuous motion planning tasks in complex environments, reinforcement learning-based navigation has been preliminarily studied recently. Compared to traditional control methods, when using reinforcement learning algorithms to address navigation problems, sufficient theoretical knowledge is not required, and the proposed model tends to solve the problem end-to-end. By defining better state space representations in complex and infinite environments, reinforcement learning algorithms
can be simplified and navigation efficiency will be improved. Jaradat et al. [227] used Q-learning to handle the problem of mobile robot navigating in an unknown dynamic environment. Owing to the infinite number of states in a dynamic environment, the authors limited the number of states based on a new definition of the state space to ensure that the navigation speed was improved. Similarly, Shi et al. [228] utilized Q-learning to predict partial missing QR codes in order to ensure image-based visual servoing. Since the QR code has a large number of feature points, the authors proposed to take its rotation and translation between the current image and the desired image as the state space to simplify the computational complexity of reinforcement learning.

During the training period, adding auxiliary tasks, such as value function [229], reward prediction [230], map reconstruction [231], and edge segmentation [232], can improve the reinforcement learning efficiency. Jaderberg et al. [230] proposed a novel unsupervised reinforcement and auxiliary learning algorithm. The algorithm predicted and controlled the features of the sensorimotor stream by treating them as pseudo-rewards for reinforcement learning. Moreover, during the training process, the agent was allowed to perform additional tasks, such as pixel control, reward prediction, and value function replay. In [231], the agent only used the visual information (images of the monocular camera) for navigation search (finding the apple in the maze). The study considered two auxiliary tasks. In the first task, a low-dimensional depth map was reconstructed at each time step, which is beneficial for obstacle avoidance and short-term path planning. The other task involved loop closure detection, wherein the agent learned to detect whether the current location had been visited within the currently running trajectory. The experiments in these studies proved that co-training can significantly improve the learning speed and performance of the model.

Recently, multimodal reinforcement learning has become a hot point and cutting edge, which combines multimodal information, such as language and video, with vision as inputs in the reinforcement learning model. To deal with navigation issues, visual language navigation (VLN) [233] has been widely used in recent years. VLN is a task that guides the embedded agent to execute natural language instructions in order to ensure image-based visual servoing. Since the QR code has a large number of feature points, the authors proposed to take its rotation and translation between the current image and the desired image as the state space to simplify the computational complexity of reinforcement learning.

During the training period, adding auxiliary tasks, such as value function [229], reward prediction [230], map reconstruction [231], and edge segmentation [232], can improve the reinforcement learning efficiency. Jaderberg et al. [230] proposed a novel unsupervised reinforcement and auxiliary learning algorithm. The algorithm predicted and controlled the features of the sensorimotor stream by treating them as pseudo-rewards for reinforcement learning. Moreover, during the training process, the agent was allowed to perform additional tasks, such as pixel control, reward prediction, and value function replay. In [231], the agent only used the visual information (images of the monocular camera) for navigation search (finding the apple in the maze). The study considered two auxiliary tasks. In the first task, a low-dimensional depth map was reconstructed at each time step, which is beneficial for obstacle avoidance and short-term path planning. The other task involved loop closure detection, wherein the agent learned to detect whether the current location had been visited within the currently running trajectory. The experiments in these studies proved that co-training can significantly improve the learning speed and performance of the model.

Recently, multimodal reinforcement learning has become a hot point and cutting edge, which combines multimodal information, such as language and video, with vision as inputs in the reinforcement learning model. To deal with navigation issues, visual language navigation (VLN) [233] has been widely used in recent years. VLN is a task that guides the embedded agent to execute natural language instructions in a 3-D environment. It requires a deep understanding of the linguistic semantics, visual perception, and most importantly, the alignment of the two. Most existing methods are based on sequence-to-sequence architecture [234]–[236]. That is, instructions are encoded as word sequences, and navigation trajectories are decoded as a series of actions, which are enhanced by attention mechanism and beam search. Therefore, connecting cross-modality training data is a key to improve training efficiency. Wang et al. [237] summarized VLN tasks and studied how to solve the three key challenges of VLN, namely cross-modal grounding, ill-posed feedback, and the generalization problems. Chaplot et al. [238] proposed a dual-attention unit to disentangle the knowledge of words in the textual representations and visual concepts in the visual representations, and align them with each other. The fixed alignment enables the learned knowledge transferred across tasks. In response to the first and second challenges, the authors proposed the reinforced cross-modal matching (RCM) method, which used reinforcement learning to connect local and global scenarios. In response to the third challenge, self-supervised imitation learning (SIL) was proposed, which helped the agent to get better policies by imitating its best performance from the past.

However, reinforcement learning-based navigation is limited to small action space and sample space, and it is generally in a discrete situation. Moreover, more complex tasks closer to the actual situation tend to have a large state space and continuous action space.

B. Deep Reinforcement Learning-Based Visual Navigation

Has achieved promising results recently by combining the perceptual ability of deep learning with the continuous decision ability of reinforcement learning. Compared to reinforcement learning-based navigation, deep reinforcement learning methods equip robots with the ability to learn high-dimensional data [239] to ensure precise perception and positioning so that they can accomplish more complex tasks, for example, navigating to different targets in a scene without retraining [240].

Deep reinforcement learning algorithms can be divided into two types: value-based and policy-based. Value-based algorithms learn the value function or the approximation of the value function, and then select a policy based on the value. Deep Q-network (DQN) is the first value-based algorithm. Tai and Liu [241] first built an exploring policy for robotics based on DQN, in order to explore a corridor environment with the depth information from an RGB-D sensor only. There are many extensions of DQN in order to improve stability and efficiency during training. Dueling DQN [242] can directly learn the value of state through the advantage function, which makes it learn faster than DQN when some of the actions do not affect the environment. On the other hand, double DQN [243] can train two Q networks at the same time and choose a smaller Q-value to reduce the overestimation error, which equips double DQN with stable performance. In that way, combining dueling DQN with double DQN is a good choice. Zeng et al. [244] utilized double Q-learning with multistep learning to handle coverage-aware UAV navigation problem. Specifically, the signal measured on the UAV was used to directly train the action-value function of the navigation policy, thus greatly maintaining the relative stability of the target and improving the learning efficiency. The original DQN can only be applied in tasks with a discrete action space. In order to extend to continuous control, many policy-based algorithms have been developed. Policy-based algorithms learn directly based on the policy without the reward. For example, deep deterministic policy gradients (DDPGs) [245] and normalized advantage function (NAF) [246] are policy-based algorithms that have been widely used. In comparison to NAF, DDPG needs less training parameters. Liu et al. [247] navigated a group of agents to provide long-term communications coverage, which only used one agent to output control decisions for all agents by employing DDPG. However, the DDPG algorithm requires researchers to spend a lot of time iterating and manually adjusting rewards in practice. To address
this problem, one way is to use some expansion of DDPG to improve sampling efficiency [248], [249]. Tai et al. [248] presented a model using asynchronous multithreading DDPG to collect data, which helped to improve sampling efficiency. The mapless motion planner could be trained end-to-end without any features designed by human or prior demonstrations. Similarly, Zhang et al. [249] proposed asynchronous episodic DDPG, which improved learning efficiency with less training time in computationally complex environments. Episodic control and a novel type of noise were introduced to the asynchronous framework in order to improve sample efficiency while increasing data throughput. Another solution is to introduce AutoRL, an evolutionary automation layer around reinforcement learning, which helps to optimize the reward and the neural network hyperparameter while learning navigation policies. Chiang et al. [250] introduced AutoRL to simultaneously train a group of agents using DDPG for several generations. Each agent had a slightly different reward function and hyperparameter to optimize the real goal-reaching the destination.

Actor-critic (AC) algorithm [251] combines two types of deep reinforcement learning algorithms mentioned above. That is, the actor network chooses the proper action in a continuous action space, while the critic network implements single-step-update, which improves the learning efficiency. In other words, it learns both the value function and the policy function. Asynchronous advantage actor-critic (A3C) network [252], an improvement of the AC network with multithreading method, as an on-policy learning algorithm, uses newly collected samples for each gradient step. A3C network performs interactive learning with the environment in multiple threads simultaneously, thereby avoiding over-fitting of the training data. When robots autonomously exploring unknown cluttered environments, A3C equips the robots with the ability to gain cross-target generalization. In order to gain cross-target generalization ability, Zhu et al. [240] took both the target and scene images as inputs of the deep reinforcement learning network; then, the agent followed the output action to navigate to a target. During the training process, a new observation was valued through the A3C network to ensure that the agent did not need to retrain the new target. Moreover, Duron et al. [253] added semantic network to the visual network proposed in [240] to learn context from the objects present in the scene. A3C network takes the features from the joint embedding layer as inputs and then outputted the next action and the Q-value for the current state. Besides, off-policy learning algorithm, such as soft actor-critic (SAC) [254], aims to reuse past experience, which provides for both sample-efficient learning and stability. de Jesus et al. [255] applied SAC to learn continuous action space policies and maximize the entropy of the policy in the mobile robotics exploration problem.

With the development in deep reinforcement learning algorithms, the problem of vanishing gradient arises. That is, as the number of hidden layers in neural networks increases, the classification accuracy in the training process decreases. The LSTM architectures [256] is a good way to tackle this problem. When the input data is time-varying, LSTM can capture the long-term dependencies of sequential data. Mnih et al. [252] used LSTM units to make better decisions by considering the previous state characteristics. In real-world navigation, training data are more variable and unpredictable than those in simulation experiments. Therefore, LSTM plays a vital role in generating good navigation policies. Mirowski et al. [257] only used the visual information as input for unmanned vehicle navigation without relying on maps, GPS, and other auxiliary tools. The authors put unmanned vehicles in complex scenes of city scale and collected real-world data for training. To accomplish the tasks, a multicity navigation network with LSTM was proposed. The method processed images, extracted features, remembered and understood the environment, and finally generated the navigation policies.

By using deep reinforcement learning methods, agents can automatically learn the characteristics of the data collected by the sensors without human intervention. On this basis, agents are able to formulate a navigation policy to ensure navigation in more complex environments, especially in real world. In the field of navigation that is biased toward obstacle avoidance, the methods used in [258] and [259] obtained satisfactory generalization performance. Therefore, the models trained solely in virtual environments are possible to be transferred to real robots. Chen et al. [258] presented a novel approach to train action policies to acquire navigation skills for wheel-legged robots using deep reinforcement learning. It is crucial that domain randomization was introduced to increase the diversity in training samples, improve the generalization ability, thereby focusing on the task-related aspects of observation. Therefore, it has been used in real environments with more complicated types of obstacles and movements. Xie et al. [259] proposed a new network structure, consisting of two parts, to deal with the obstacle avoidance problems. First, the convolutional residual network was used to extract the depth information. Then the reinforcement learning structure could efficiently learn how to avoid obstacles in a simulator even with very noisy depth information predicted from the RGB images.

To improve the performance of deep reinforcement learning networks, training data should be essentially considered in experiments. Sufficient and variable training data are the basis of convincing results during the training process, while in the real world, training data are always unobtainable or missing. To handle this problem, simulation frameworks can be utilized to train agents. For example, in [240], the first simulation framework, called AI2-The House Of inteRactions (THOR), was developed to provide an environment with high-quality 3-D scenes as well as physics engines. Therefore, the robot in a simulation environment can effectively collect several training samples, which improves the data utilization. Specifically, Wu et al. [260] analyzed the cross-target and cross-scene generalization ability of the target-driven navigation models on AI2-THOR. The evaluation, which was conducted in 120 synthetic scenes from four categories, including kitchen, living room, bedroom, and bathroom, greatly exceeded some relative baselines.

After training in simulation, it is difficult to ensure that the agent achieves similar performances between the virtual
scene and the real scene because of the domain shift. One possible solution is to add the vSLAM map in the navigation process, which helps to narrow the difference in performance between simulation and real environment. On the basis of [250], Francis et al. [261] introduced the vSLAM map to robot navigation, in order to reconstruct the motion probability map. Since the vSLAM map is noisy, it can compensate for the difference in performance between the robots in the virtual and the real environment due to the different levels of noise. From another perspective, constructing an exploration framework bridges the gap between simulation and the real environment. Li et al. [232] constructed a framework consisting of mapping, decision, and planning. Each module was independent and can be achieved by a variety of methods. Compared to traditional end-to-end deep reinforcement learning methods with raw sensor data as input and control policy as output, the proposed deep reinforcement learning algorithm based on framework learned faster and equipped itself with better generalization performance in different maps.

IV. DISCUSSION

A. Deep Learning-Based Visual Perception

The constructed map is an intuitive representation of the scene perception and the basis for intelligent robots to autonomously perform advanced tasks. Mapping has undergone a development process from 2-D to 3-D, from sparse to dense, and from topological to semantic, among others. Furthermore, although several methods have been proposed to improve the localization accuracy, there are still many challenges remaining to be solved. Therefore, we summarize the challenges and promising directions of perception as follows.

1) Accurate Perception: Although learning-based perception algorithms have made great progress in the perception areas, their accuracy, especially the accuracy of unsupervised learning methods, still has much room for improvement. Digging the more effective constraints for training from the aspect of geometry, cross-task relationships, and interpretability, utilizing novel learning frameworks, like meta-learning, curriculum learning, and lifelong learning to make full use of the data, and developing more efficient neural network frameworks for feature extraction and inference are both promising directions.

2) Robust Perception: Robustness is one of the most important indicators for the reality application of perception algorithms. Although current learning systems have received good accuracy on the datasets, the network will be affected by the sensor noise, lighting, and scenarios when being used to real environments. Therefore, the robust environmental perception, ego-motion perception, and navigation based on learning systems under different conditions (like different seasons, different weather, different lighting conditions, different source sensors, indoor and outdoor as well as day and night) in the same scene are problems to be handled.

3) Real-Time Perception: Real-time perception is important for autonomous systems in practical applications. Current high accuracy networks are based on complex network structures, which includes a huge number of parameters and large Flops. Therefore, the training and application of deep neural networks have a higher demand on the computing power of the systems, which limits the practical applications. Using novel lightweight learning architectures, such as lightweight network and knowledge distillation, to improve the real-time performance of perception networks will be another trend.

4) Geometry Assist in Perception: Utilizing the geometric prior built by a learning framework or knowledge graph in the perception of autonomous systems is helpful and a promising direction with broad development prospects. For example, semantic labels predicted by deep learning are used to correlate with the knowledge graph of objects to obtain prior geometric information, such as the size of objects; therefore, the detailed scale, structure, and 3-D information can be obtained.

5) Representation of the Environment Based on Deep Learning: Representing the environment based on deep learning is another challenge and a promising direction. Although previous works such as [169], [171], [177] leveraged the deep learning into mapping, the maps of these methods are still built traditionally. With the developments of Nerf algorithms [262], [263], it provides a way to present the scene by using neural networks. Most recent work has tried to construct the SLAM systems based on Nerf [264], and this is quite an interesting and promising direction.

6) Multisensor Data Fusion Based on Deep Learning: Fusing information from multisensors (IMU, LiDAR, event-based camera, or infrared camera ) or multiagent is an effective way to deal with poor quality input images comprising motion blur and recover scale information. However, expressing the additional sensor information explicitly in the constraints for training is a significant challenge. For example, the current methods leverage IMU data with images for pose estimation in a supervised manner [139], [140], and the information from IMU is not represented in the loss function. Thus, whether the IMU data play an important role in pose estimation and what role it plays is unknown and not yet explainable.

7) Integration of Deep Learning and Traditional Frameworks: Although a lot of relevant research has been summarized above, there is still a lot of work to be done in this direction. With the help of deep learning, the basic idea is to improve the traditional frameworks by analyzing the limitations and shortcomings of traditional methods. For example, considering that the current direct methods rely heavily on the photometric consistency assumption, we can use deep learning to perform a photometric correction or transform images into photometric-consistent feature maps.

B. Reinforcement Learning-Based Visual Navigation

There is still a long way to go before reinforcement learning can be applied to autonomous systems. Therefore, there are many challenges to be addressed.
1) **Sparse Rewards:** Rewards have a great impact on the learning results during the training process, but the problem of sparse rewards in reinforcement learning has not been well solved. When the training tasks are complicated, the probability of exploring the target (getting positive rewards) by random exploring becomes very low. Therefore, it is difficult for reinforcement learning algorithms to converge by only relying on the positive rewards. To deal with this problem, redesigning the reward function according to specific scenarios will be helpful in avoiding the problem of sparse rewards, and by means of hierarchical networks, such as hierarchical reinforcement learning [265], the training efficiency, and the final performance will be improved.

2) **Complicated Calculation and High Cost:** When a robot navigates in a large-scale or continuous state space environment, the calculation process is likely to be high-dimensional and complex. Moreover, reinforcement learning algorithms require thousands of trials and errors to train iteratively, while in real-world tasks, agents hardly withstand so many trials and errors because of limited cost. Therefore, the efficient feature representation of large-scale space reinforcement learning will simplify the calculation process. In other words, the use of efficient and fast online self-evaluation reinforcement learning algorithms will reduce the training costs and improve learning efficiency.

3) **Performance Between Simulation and Real World:** Due to the huge gap between simulation environments and real scenes, many reinforcement learning algorithms with higher performance in simulation experiments cannot handle the practical problems in the real world, which strongly limits the widespread application of this technology. Establishing a network that can be directly transferred to the real world and building a high-fidelity simulation and physical platform will be a future trend, which can effectively convert virtual scenes generated in a simulator into real scenes for reinforcement learning training.

4) **Transferable Property Improvement:** In many tasks, the training data is limited and unobtainable. Adversarial learning methods [266] can be applied to increase the data differences in the training process and reduce data differences in the testing process, which improves the data diversity and the generalization ability of the model. Moreover, many transfer learning methods, such as few-shot learning, zero-shot learning, and meta learning [267]–[269], can recognize the new model and apply the knowledge and skills learned in previous tasks to novel tasks with few or even no training data. This is effective for enhancing the transferability, reducing network parameters, and promoting generalization.

5) **Multimodal and Multitask:** Current reinforcement learning-based navigation methods mainly focus on visual input. However, by considering the information from multiple models, such as voice, text, and video, the agents can better understand the scenes and the performance in experiments will be more accurate and efficient. Moreover, it is proven that multitask reinforcement learning models [270], in which the agent is simultaneously trained with auxiliary and target tasks, improve the training efficiency. Therefore, multimodal and multitask are also development trends in navigation based on reinforcement learning.

### C. Application

The developments of autonomous environment perception and motion planning drive the emergence of a large number of high-tech industries, such as unmanned vehicles and service robots, which have greatly improved the quality of human life [271]. Furthermore, autonomous systems have a broad application prospect in various fields, like industry, agriculture, services, and transportation. For example, accidents in petrochemical industry have occurred from time to time in recent years, which inevitably caused great damage of life and property. The use of autonomous systems monitoring a chemical park can help to find dangers in advance. Intelligent monitoring robots with various gas and optical sensors can monitor the safety hazards in the chemical plant area in real-time. Robots autonomously perceive and construct the map of structure environments based on visual sensors. Then, based on the perceived information, robots plan the path and tasks for better monitoring. In case of emergency, the environment becomes semi-structured and complicated, in which autonomous robots can reach dangerous areas, sense the surrounding areas, deliver important information to the staff, assess the situation, and even assist staff in decision-making as well as rescue. At present, the difficulties lie in the distance between theoretical research and practical applications, such as reliability, robustness, and real-time response capability. Therefore, this survey reviews the existing learning-based perception and navigation methods, which provide a guideline for future research and promote the developments of autonomous systems.

### V. Conclusion

Through this review, we aim to contribute to this growing area of research by exploring the learning-related methods for self-state perception, environment perception, and navigation in autonomous systems. Therefore, we review the related works of learning-based vSLAM and navigation in the learning age. The influx of deep learning algorithms can be observed to support the subtasks of vSLAM or incorporate with vSLAM in recent works, which improve the robustness and performance of traditional vSLAM algorithms. Meanwhile, navigation based on deep reinforcement learning achieves good efficiency and transferability in autonomous systems. We provide two comprehensive taxonomy tables of state-of-the-art vSLAM algorithms as well as deep learning-based depth and pose estimation methods, which clarify the mainstream algorithm framework and the development trend. Finally, this review highlights the key challenges and promising directions in learning-based perception and navigation.
Q. Sun, Y. Zhang, Z. Chen, Y. Fu, and X. Xue, “Depth-guided AdaIN and shift attention network for vision-and-language navigation,” in Proc. IEEE Int. Conf. Multimedia Expo. (ICME), Jul. 2021, pp. 1–6.

X. Wang et al., “Reinforced cross-modal matching and self-supervised imitation learning for vision-language navigation,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 6661–6668.

D. S. Chaplot, L. Lee, R. Salahkhutdinov, D. Parikh, and D. Batra, “Embodied multimodal multitask learning,” in Proc. 29th Int. Joint Conf. Artif. Intell., Jul. 2020, pp. 2442–2448.

H. Lin, S. Garg, J. Hu, G. Kaddoum, M. Peng, and M. S. Hossain, “Blockchain and deep reinforcement learning empowered spatial crowdsourcing in software-defined internet of vehicles,” IEEE Trans. Intell. Transp. Syst., vol. 22, no. 6, pp. 3755–3764, Jun. 2021.

Y. Zhu et al., “Target-driven visual navigation in indoor scenes using deep reinforcement learning,” in Proc. IEEE Int. Conf. Robot. Autom. (ICRA), May 2017, pp. 3357–3364.

L. Tai and M. Liu, “A robot exploration strategy based on Q-learning network,” in Proc. IEEE Int. Conf. Real-time Comput. Robot. (ICCAR), Jun. 2016, pp. 57–62.

Z. Wang, T. Schaul, M. Hessel, H. Hasselt, M. Lanctot, and N. Freitas, “Dueling network architectures for deep reinforcement learning,” in Proc. Int. Conf. Mach. Learn., 2016, pp. 1995–2003.

H. Van Hasselt, A. Guez, and D. Silver, “Deep reinforcement learning with double Q-learning,” in Proc. AAAI Conf. Artif. Intell., vol. 30, no. 1, 2016, pp. 2094–2100.

Y. Zeng, X. Xu, S. Jin, and R. Zhang, “Simultaneous navigation and radio mapping for cellular-connected UAV with deep reinforcement learning,” IEEE Trans. Wireless Commun., vol. 20, no. 7, pp. 4205–4220, Jul. 2021.

T. P. Lillicrap et al., “Continuous control with deep reinforcement learning,” 2015, arXiv:1509.02971.

S. Gu, T. Lillicrap, I. Sutskever, and S. Levine, “Continuous deep Q-learning with model-based acceleration,” in Proc. Int. Conf. Mach. Learn., 2016, pp. 2829–2838.

C. H. Liu, Z. Chen, J. Tang, J. Xu, and C. Piao, “Energy-efficient UAV control for effective and fair communication coverage: A deep reinforcement learning approach,” IEEE J. Sel. Areas Commun., vol. 36, no. 9, pp. 2059–2070, Sep. 2018.

L. Tai, G. Paolo, and M. Liu, “Virtual-to-real deep reinforcement learning: Continuous control of mobile robots for mapless navigation,” in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS), Sep. 2017, pp. 31–36.

Z. Zhang, J. Chen, Z. Chen, and W. Li, “Asynchronous episodic deep deterministic policy gradient: Toward continuous control in computationally complex environments,” IEEE Trans. Cybern., vol. 51, no. 2, pp. 604–613, Feb. 2021.

H.-T.-L. Chiang, A. Faust, M. Fiser, and A. Francis, “Learning navigation behavior end-to-end with AUTORL,” IEEE Robot. Autom. Lett., vol. 4, no. 2, pp. 2007–2014, Apr. 2019.

V. R. Konda and J. N. Tsitsiklis, “Actor-critic algorithms,” in Handbook of Dynamic Programming and Decision Processes, New York: Wiley, 2000, pp. 1008–1014.

V. Mnih et al., “Asynchronous methods for deep reinforcement learning,” in Proc. Int. Conf. Mach. Learn., 2016, pp. 1928–1937.

R. Druon, Y. Yoshiyama, A. Kanazaki, and A. Watt, “Visual object search by learning spatial context,” IEEE Robot. Autom. Lett., vol. 5, no. 2, pp. 1279–1286, Apr. 2020.

T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, “Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor,” in Proc. Int. Conf. Mach. Learn., 2018, pp. 1861–1870.

J. C. de Jesus, V. A. Kich, A. H. Kolling, R. B. Grando, M. A. D. S. L. Cuadros, and D. F. T. Gamarra, “Soft actor-critic for navigation of mobile robots,” J. Intell. Robotic Syst., vol. 102, no. 2, pp. 1–11, Jun. 2021.

T.-Y. Lee, J. van Baar, K. Wittenburg, and A. Sullivan, “Analysis of the contribution and temporal dependency of LSTM layers for reinforcement learning tasks,” in Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. Workshops, Jan. 2019, pp. 99–102.

P. Mirowski et al., “Learning to navigate in cities without a map,” in Proc. Adv. Neural Inf. Process. Syst., 2018, pp. 2419–2430.

X. Chen, A. Ghadirzadeh, J. Folkesson, M. Bjorkman, and P. Jensfelt, “Deep reinforcement learning to acquire navigation skills for wheellegged robots in complex environments,” in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS), Oct. 2018, pp. 3110–3116.

L. Xie, S. Wang, A. Markham, and N. Trigoni, “Towards monocular vision based obstacle avoidance through deep reinforcement learning,” 2017, arXiv:1706.09829.
Jianrui Wang received the B.S. degree in automation from Dalian Maritime University, Dalian, China, in 2019. He is currently pursuing the Ph.D. degree in control science and engineering with the School of Information Science and Engineering, East China University of Science and Technology, Shanghai, China. His current research interests include deep reinforcement learning, multiagent reinforcement learning, game theory, and their applications.

Chongzheng Zhang received the B.S. degree from Nanjing Forestry University, Nanjing, China, in 2018, and the master’s degree from the East China University of Science and Technology, Shanghai, China, in 2021. She is currently a Project Manager with the Shanghai AI Laboratory, Shanghai.

Qiyu Sun received the B.S. degree in automation from the East China University of Science and Technology, Shanghai, China, in 2019, where she is currently pursuing the Ph.D. degree in control science and engineering. Her research interests include 3-D scene understanding, domain adaptation, and deep learning.

Wei Xing Zheng (Fellow, IEEE) received the B.Sc. degree in applied mathematics, the M.Sc., and Ph.D. degrees in electrical engineering from Southeast University, Nanjing, China, in 1982, 1984, and 1989, respectively. Over the years, he has held various faculty/research/visiting positions at Southeast University, Nanjing, China; the Imperial College of Science, Technology and Medicine, London, U.K.; The University of Western Australia, Perth, WA, Australia; the Curtin University of Technology, Perth; the Munich University of Technology, Munich, Germany; the University of Virginia, Charlottesville, VA, USA; and the University of California at Davis, Davis, CA, USA. He is currently a University Distinguished Professor with the Western Sydney University, Sydney, NSW, Australia. Dr. Zheng was named a Highly Cited Researcher by Clarivate Analytics (formerly Thomson Reuters) in 2015, 2016, 2017, 2018, 2019, 2020, and 2021, consecutively. He was a recipient of the 2017 Vice-Chancellor’s Award for Excellence in Research (Researcher of the Year) at Western Sydney University. He served as the Chair for IEEE Circuits and Systems Society’s Technical Committee on Neural Systems and Applications and the IEEE Circuits and Systems Society’s Technical Committee on Blind Signal Processing. He is also a Distinguished Lecturer of IEEE Control Systems Society, IEEE Transactions on Automatic Control, IEEE Signal Processing Letters, IEEE Transactions on Circuits and Systems—II: Express Briefs, IEEE Transactions on Fuzzy Systems, and IEEE Transactions on Neural Networks and Learning Systems. He has also served as a Guest Editor for IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II: EXPRESS BRIEFS, IEEE TRANSACTIONS ON FUZZY SYSTEMS, and IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS. He is currently the Editor-in-Chief of CHAOS, an Associate Editor of IEEE TRANSACTIONS ON NETWORKS AND COMMUNICATIONS, and an Associate Editor of IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS. He is also the Editor-in-Chief of CHAOS.

Feng Qian (Member, IEEE) received the B.Sc. degree in chemical automation and meters from the Nanjing Institute of Chemical Technology, Nanjing, China, in 1982, and the M.Sc. and Ph.D. degrees in automation from the East China Institute of Chemical Technology, Shanghai, China, in 1988 and 1995, respectively. He was the Director of the Automation Institute, East China University of Science and Technology, from 1999 to 2001, and the Head of the Scientific and Technical Department from 2001 to 2006. He is currently the Vice President of the East China University of Science and Technology, and the Director of the Key Laboratory of Smart Manufacturing in Energy Chemical Process, Ministry of Education, Shanghai, and the Process System Engineering Research Center, Ministry of Education, Shanghai. His current research interests include modeling, control, optimization, and integration of petrochemical complex industrial processes and their industrial applications, neural network theory, and real-time intelligent control technology and its applications to the ethylene, PTA, PET, and refining industries.

Jürgen Kurths received the B.S. degree in mathematics from the University of Rostock, Rostock, Germany, in 1975, the Ph.D. degree from the Academy of Sciences of the German Democratic Republic, Berlin, Germany, in 1983. The Honorary degree from the N. I. Lobachovsky State University of Nizhny Novgorod, Russia, in 2008, and the Honorary degree from Saratov State University, Russia, in 2012. He was a Full Professor with the University of Potsdam, Potsdam, Germany, from 1994 to 2008. Since 2008, he has been a Professor of nonlinear dynamics with the Humboldt University of Berlin, Berlin, and the Chair of the research domain transdisciplinary concepts with the Potsdam Institute for Climate Impact Research, Potsdam. Since 2009, he has been the Sixth-Century Chair with the University of Aberdeen, Aberdeen, U.K. He has authored over 680 articles, which are cited more than 40 000 times (H-index: 104). His current research interests include synchronization, complex networks, time series analysis, and their applications. Dr. Kurths became a member of the Academia Europaea in 2010, the Macedonian Academy of Sciences and Arts in 2012, and the Royal Society of Edinburgh in 2021. He is currently a fellow of the American Physical Society. He received the Alexander von Humboldt Research Award from the Council of Scientific and International Research, India, in 2015. He is named as an ISI Highly Cited Researcher in physics and engineering by Thomson Reuters. He is also the Editor-in-Chief of CHAOS.