Cognitive Computation (2018) 10:875–889
https://doi.org/10.1007/s12559-018-9591-8

Ongoing Evolution of Visual SLAM from Geometry to Deep Learning: Challenges and Opportunities

Ruihao Li[1] · **Sen Wang**[2] · **Dongbing Gu**[1]

Received: 4 April 2018 / Accepted: 29 August 2018 / Published online: 8 September 2018

Abstract

Visual simultaneous localization and mapping (SLAM) has been investigated in the robotics community for decades. Significant progress and achievements on visual SLAM have been made, with geometric model-based techniques becoming increasingly mature and accurate. However, they tend to be fragile under challenging environments. Recently, there is a trend to develop data-driven approaches, e.g., deep learning, for visual SLAM problems with more robust performance. This paper aims to witness the ongoing evolution of visual SLAM techniques from geometric model-based to data-driven approaches by providing a comprehensive technical review. Our contribution is not only just a compilation of state-of-the-art end-to-end deep learning SLAM work, but also an insight into the underlying mechanism of deep learning SLAM. For such a purpose, we provide a concise overview of geometric model-based approaches first. Next, we identify visual depth estimation using deep learning is a starting point of the evolution. It is from depth estimation that ego-motion or pose estimation techniques using deep learning flourish rapidly. In addition, we strive to link semantic segmentation using deep learning with emergent semantic SLAM techniques to shed light on simultaneous estimation of ego-motion and high-level understanding. Finally, we visualize some further opportunities in this research direction.

Keywords SLAM · Deep learning · Depth estimation · Pose estimation · Semantic mapping

Introduction

Visual simultaneous localization and mapping (SLAM) is essential to achieve persistent autonomy for vision-based mobile robots, especially in unknown environments. It is also a key enabler for enormous vision-based applications, such as virtual and augmented reality. Researchers from the robotics and computer vision communities have endeavored and managed to design some efficient and versatile visual SLAM systems in the past several decades.

✉ Ruihao Li
rlig@essex.ac.uk

Sen Wang
s.wang@hw.ac.uk

Dongbing Gu
dgu@essex.ac.uk

[1] School of Computer Science and Electronic Engineering, University of Essex, Colchester, CO4 3SQ, UK

[2] Edinburgh Centre for Robotics, Heriot-Watt University, Edinburgh, EH14 4AS, UK

Most of the existing visual SLAM methods explicitly model camera projections, motions, and environments based on visual geometry. Therefore, they are referred to as model-based SLAM. They can be divided into feature-based methods [1–3] and direct methods [4–6] according to the means that image information is used. Specifically, feature-based visual SLAM methods extract sparse features, such as points and lines, from the images for feature matching and ego-motion estimation, while direct methods directly use dense (or semi-dense) image pixels for motion estimation under the assumption of photometric consistency. Loop-closure detection and back-end optimization can be incorporated with both methods to form a full visual SLAM system.

The state-of-the-art model-based visual SLAM algorithms have made a great success in the past decade. Superior performance on localization and mapping accuracy, for example, has been demonstrated by both feature-based [3] and direct [6] methods. However, they still face many challenging issues, in particular when being deployed in large-scale environments, or under extreme lighting conditions. Nowadays system robustness [7] and high-level (semantic)

perception [7, 8] are the demanding tasks for visual SLAM systems. Unfortunately, it becomes increasingly challenging to solve these problems by solely relying on model-based methods. One of the reasons is that the high-dimensional images carry significant "redundant" information and the real world has complex appearance, which is difficult to be manually modeled in a precise manner.

Deep learning can automatically learn effective feature representations from massive data in an end-to-end fashion, and do not need the extraction of manually designed features [9]. In this way, deep learning can learn more robust and effective features according to the specific problems, and has successfully demonstrated the good capability for some challenging cognitive and perceptual tasks, such as handwritten code recognition [10], human pose estimation[11], tactile recognition [12], and facial landmark localization [13]. Unavoidably, the evolution of visual SLAM from model-based methods to deep learning methods occurs. Recent attempts to develop deep learning approaches for visual SLAM problems include the depth estimation of a scene from a monocular image [14], the visual odometry estimation [15], and the semantic map generation [8]. These recent advances promise a huge potential for visual SLAM systems to address the challenging issues by bringing in adaptive and learning capability.

This paper is to provide a review of the ongoing change of visual SLAM systems from model-based to deep learning methods. There are some previous publications which provide various overviews of SLAM techniques. Durrant-Whyte et al. [16, 17] presented a tutorial on the essential methods for solving the SLAM problem from a view of the recursive Bayesian formulation. Scaramuzza et al. [18, 19] provided a comprehensive review of visual odometry (VO). Cadena et al. [7] provided a detailed survey on visual SLAM and delineated some open challenges and research issues, including system robustness and semantic perception. However, they mainly concentrate on model-based methods with no or limited discussion on data-driven approaches. Technically they focus on the selection of features, the framework of recursive optimizations, or the detection of loop closures. Clearly, our work is distinct from them by focusing on the recent advances of visual SLAM methods using deep learning, covering the construction of deep neural networks, the design of loss functions and the flexibility of estimations. Further, we also unfold how deep learning SLAM can benefit from model-based methods when a loss function is designed or when a deep learning architecture is constructed. We also emphasize how important semantic SLAM can be given the strong capability of deep learning for image segmentation. Finally, future opportunities in this direction are focused on system robustness, semantic understanding, and learning capability.

The rest of this paper is organized as follows. "Model-based SLAM Methods" reviews related works of model-based visual SLAM methods. Three deep learning networks and available training data sets are given in "Deep Neural Networks for Visual SLAM." Depth estimation methods with deep learning are surveyed in "Depth Estimation with Deep Learning," followed by the review of pose estimation methods with deep learning in "Pose Estimation with Deep Learning". "Semantic Mapping with Deep Learning" introduces the state-of-the-art semantic mapping methods with deep learning. Finally, "Open Challenges and Future Opportunities" proposes some potential opportunities for deep learning SLAM, before drawing the conclusion in "Conclusions."

Model-based SLAM Methods

Model-based SLAM methods explicitly model camera projections, motions, and environments based on multi-view geometry and photometric consistency. They can be divided into feature-based methods and direct methods. Feature-based methods extract and match feature points from 2D images, and then compute and optimize camera poses along with the positions of these feature points in 3D. In contrast, direct methods use pixels in the image to compute 6-DoF camera poses directly by minimizing photometric errors without extracting feature points. Although there exist plenty of model-based methods, we mostly focus on the state-of-the-art in terms of localization and mapping accuracy, due to the space limitation.

Feature-Based Visual SLAM Methods

MonoSLAM [20] proposed by Davison et al. is one of the earliest real-time visual SLAM systems with a monocular camera. Different from Structure from Motion (SfM) approaches that are lack of real-time performance, MonoSLAM adopts a probability framework and creates sparse yet consistent 3D feature points for a map. By combining a general camera motion model and feature initialization, MonoSLAM achieves 3D localization and mapping with 30 Hz real-time performance on a standard PC. MonoSLAM bridges pure vision and autonomous robotics closer and provides some new potential applications for augmented reality (AR).

However, tracking and mapping in MonoSLAM system are intimately linked and operated in one single thread. In other words, the 6-DoF camera pose and the 3D map points are updated together at every frame. The algorithm could only handle a limited number of sparse features due to the use of the large volume of images. In order to tackle this problem, Klein et al. proposed a parallel tracking and

mapping (PTAM) system [1] which separates tracking and mapping into two parallel threads. The mapping thread is updated according to keyframes and is performed using computationally expensive bundle adjustment technology. The tracking thread is updated at frame rate to estimate 6-DoF camera poses based on the built 3D map. PTAM is successfully performed with a hand-held camera in a small environment.

ORB-SLAM proposed by Mur-Artal et al. [3] is one of the most successful feature-based SLAM systems by now. They first proposed a place recognition system [21] with ORB features based on Bag-of-Words (BoW) technology. ORB [22] is a rotational invariant and scale aware feature, and could be extracted in high frequency. The proposed place recognition algorithm can run efficiently, resulting in the real-time implementation of both relocalization and loop closing in visual SLAM systems. Then building upon the ORB place recognizer [21], they proposed ORB-SLAM [3] with monocular cameras, which could be performed in large-scale environments and has demonstrated a superior performance. Afterward, they extended ORB-SLAM from monocular cameras to stereo and RGB-D cameras [23].

Endres et al. proposed RGB-D SLAM [2], which is based on feature points. The proposed RGB-D SLAM can generate dense and accurate 3D maps. In recent years, there appears a new kind of sensor called event camera or Dynamic and Active-pixel Vision Sensor (DAVIS) . The corresponding SLAM algorithms [24] [25] are proposed for 6-DoF motion tracking and 3D reconstruction, and these algorithms demonstrate impressive performance in some challenging scenarios.

The shift from low-level point features to high-level objects is also observed in emerging semantic SLAM. Salas-Moreno et al. [26] presented a planar SLAM system which can detect the planar in environments and yield a planar map. They also proposed a SLAM system called SLAM++ [27], which can detect objects, such as chairs and desks, and then utilize these objects for localization. However, a limited number of objects, such as planar, desks, and chairs, are extracted and specific supervised off-line learning is required.

Direct Visual SLAM Methods

Different from the above feature-based methods, direct methods do not rely on manually designed sparse features. They instead employ most pixels in an image to estimate 6-DoF camera poses by penalizing some photometric errors for each overlapping image pair.

Newcombe et al. proposed a dense tracking and mapping (DTAM) system [4]. As the depth of each pixel in an image is estimated, DTAM generates a dense 3D map for each frame. Afterward, Newcombe et al. [28] proposed KinectFusion using a RGB-D camera which is successfully demonstrated in dense registration and mapping. KinectFusion relies on a truncated signed distance function (TSDF) for pixel grid representation and utilizes iterative closest point (ICP) for aligning depth images. Both DTAM and KinectFusion run in room scale environments with commercial GPU for real-time performance.

Whelan et al. presented ElasticFusion [29] based on surfel representation with a RGB-D camera. By using frame-to-model tracking and non-rigid deformation, ElasticFusion performs the time-windowed surfel-based dense data fusion. The dense global consistent map is obtained without the need for pose graph optimization or post-processing steps. GPU is also required for camera tracking and dense mapping in order to achieve real-time performance.

In order to increase the efficiency of dense-based methods, Engel et al. proposed semi-dense visual odometry (SVO) [30] which runs real-time on CPU. SVO uses the pixels with a non-negligible image gradient rather than all the pixels in the image. The semi-dense inverse depth map is estimated and 6-DoF camera motion is tracked with the alignment of estimated depth maps. Forster et al. also presented a similar approach called SVO [5]. Engel et al. then improved SVO [30] by introducing Large-Scale Direct Monocular SLAM (LSD-SLAM) [31] which can run in large-scale environments with CPU. LSD-SLAM employs $\mathfrak{sim}(3)$ to detect scale drifts and provides a probabilistic solution to handle the noisy depth prediction during tracking. Recently, Engel et al. further improved the direct method and proposed Direct Sparse Odometry (DSO) [6]. DSO combines photometric errors with geometric errors and optimizes all the model parameters jointly. The demonstrated performance includes high accuracy in tracking and mapping, and robustness in some featureless environments.

Pascoe et al. proposed NID-SLAM [32] which is also a direct method for monocular cameras. Instead of penalizing photometric errors like most direct methods, NID-SLAM chooses normalized information distance (NID) metric to estimate the camera motion. NID-SLAM demonstrates robust performance in appearance changing environments.

Summary

Model-based visual SLAM methods have successfully demonstrated their superior capabilities in pose estimation and 3D map construction. Particularly the feature-based representative ORB-SLAM [3] and the direct representative DSO [6] both achieve high accuracy in the large-scale environment, and real-time performance with commercial CPUs. However, their robustness still tends to be struggling when they face some featureless environments or other

challenging scenes, e.g, serious image blur. Further, they do not have a learning capability to be adaptive to specific circumstances. The success of deep learning in computer vision sheds some light on the improvement of robust performance through the continuous learning.

Deep Neural Networks for Visual SLAM

Model-based methods represent the input images with manually designed features and search for the best pose which matches the features between image frames. Deep learning directly learns good representations of the input images at multiple levels. The representation could be unknown features, depth, or even ego-motion between two frames for SLAM problems.

In this section, three types of deep neural networks (DNNs) are briefly illustrated, which have been already found in deep learning SLAM methods. For more information on deep learning, the reader is referred to [9].

Convolutional Neural Network

Convolutional neural network (CNN) is one of the most popular deep neural network architectures to date. A CNN mainly consists of vision layers (e.g., convolutional layer, activation layer, pooling layer) and common layers (e.g., fully connected layer), as shown in Fig. 1a. Dropout layers and normalization layers (e.g., batch normalization layer) are also frequently incorporated. Loss function, such as Softmax and Euclidean loss, drives the training by minimizing the differences between the predictions and the labels.

Convolutional layers are a core component of CNNs although the specific structures of CNNs vary from case to case. Given an input tensor $\mathbf{x}$, they produce an output feature map $\mathbf{h}^k$ by the convolution of an input image with a linear filter $\mathbf{W}^k$:

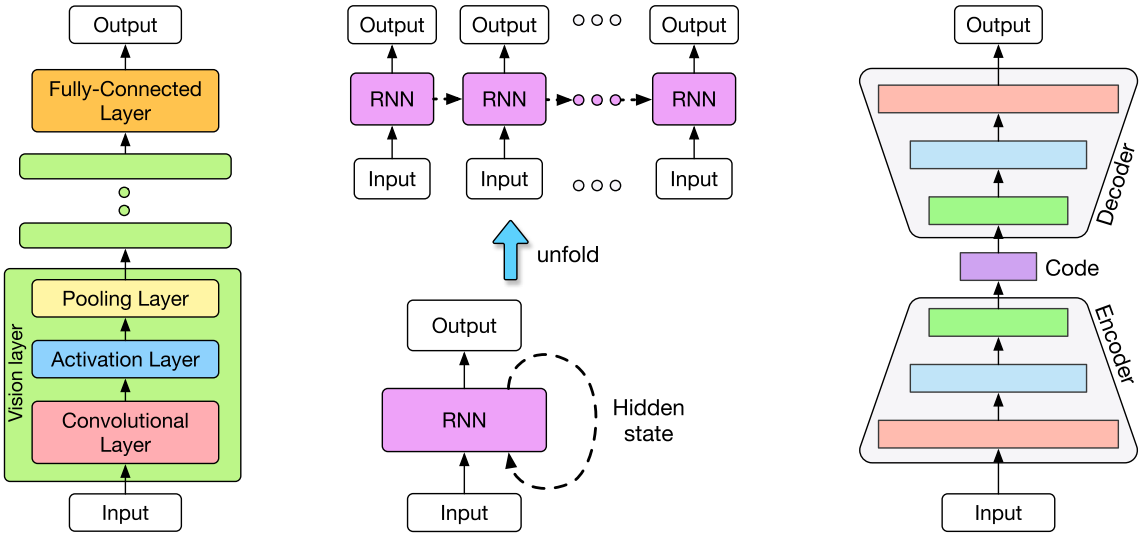
$$\mathbf{h}_{ij}^k = (\mathbf{W}^k * \mathbf{x})_{ij} + \mathbf{b}_k \tag{1}$$

where $\mathbf{b}_k$ is a bias term. Thanks to the local connectivity and parameter sharing of the convolutional layers, CNNs vastly reduce the number of parameters and are more efficient than multi-layer perceptron.

The loss function is a vital part of DNNs. With a carefully designed loss, the network can learn how to solve different kinds of problems. Cross-entropy loss is usually used for classification problems, while the Euclidean loss is mainly employed for regression ones. Kendall [33] used the Euclidean loss to compute differences of pose prediction and ground truth to solve pose regression problems.

Deep Recurrent Neural Network

Recurrent neural network (RNN) is mainly designed to capture the temporal dynamics of video clips. It maintains the memory of its hidden states over time via feedback loops, and models the dependencies between current input and previous states. The RNN and its unfolded version are



(a) CNN. (b) RNN. (c) Auto-Encoder.

Fig. 1 Three popular deep learning architectures: convolutional neural network, recurrent neural network, and autoencoder

shown in Fig. 1b. Given an input $\mathbf{x}_k$ at time k, a simple RNN updates at time k by:

$$\mathbf{h}_k = H(\mathbf{W}_{xh}\mathbf{x}_k + \mathbf{W}_{hh}\mathbf{h}_{k-1} + \mathbf{b}_h) \quad (2)$$

$$\mathbf{y}_k = \mathbf{W}_{hy}\mathbf{h}_k + \mathbf{b}_y \quad (3)$$

where $\mathbf{h}_k$ and $\mathbf{y}_k$ are the hidden state and output at time k, respectively. $\mathbf{W}$ is the weight matrix, $\mathbf{b}$ is the bias term, and $H()$ is a non-linear function. However, simple RNN suffers from the vanishing gradient problem in practice. In order to solve this problem, long short-term memory (LSTM) is widely used. Specifically, an LSTM has several gates to control when to keep or forget the memory. In deep RNNs related to visual SLAM, the RNNs are usually connected to the features of CNNs. This forms a paradigm termed recurrent convolutional neural network (RCNN), in which CNNs and RNNs capture the spatial and temporal representations from video clips, respectively. Therefore, RCNN is suitable to be used for solving pose estimation in SLAM problems for video clips.

Autoencoder

Autoencoder is a special kind of DNNs derived from CNN. As shown in Fig. 1c, it consists of an encoder part and a decoder part. Specifically, an autoencoder maps its input $\mathbf{x}$ into a hidden code $\mathbf{y}$ through the encoder part:

$$\mathbf{y} = e(\mathbf{x}) \quad (4)$$

where $e()$ is a non-linear function representing the encoder. Then the decoder part maps the hidden code $\mathbf{y}$ into a reconstruction $\mathbf{z}$ that usually represents the same main features with input $\mathbf{x}$. Its map function is:

$$\mathbf{z} = d(\mathbf{y}) \quad (5)$$

where $d()$ is a non-linear function denoting the decoder network. For the decoder part, deconvolution layers, dilated convolution layers, upsampling, and convolution layers are often used for feature decoding. For the autoencoder, the output image and the input image usually have the same size. And since Long et al. proposed the fully connected network (FCN) [34], Autoencoder has been widely used for depth estimation and semantic segmentation. We will discuss the details in the following sections.

Dataset

Deep learning methods require a large amount of data for training. In this part, we review the existing datasets which could be adopted for deep learning related visual SLAM tasks.

The KITTI benchmark [35] was collected in outdoor environments with a driving car. It provides stereo images with ground-truth 6-DoF poses derived from the fusion of multiple sensor data. Depth data is also provided with the calibrated laser. Some images in KITTI are labeled manually for image segmentation. More details can be seen in [36]. Similar to KITTI, Cityscapes dataset [37] also has stereo images, depth images, semantically labeled images, and 6-DoF poses. The RobotCar dataset [38] was collected with a car driving in Oxford for a year, which means it contains different weather and sceneries of a same place. EuRoc MAV dataset [39] was gathered by using a flying robot and could be used for VO and SLAM problems. TUM dataset [40] and NYU dataset [41] were collected with a hand-held RGB-D camera in indoor environments. They provide color and depth images. In addition, NYU dataset [41] also provides some labeled images for semantic segmentation. PASCAL VOC [42], the Synthia dataset [43], and COCO [44] datasets with labeled images aim at image segmentation problems. ADE20K [45] contains more than 20K pixel-wise semantic annotated images.

To summarize, KITTI [35], TUM [40], and NYU [41] datasets can be used for depth estimation. For relocalization problem, one can use 7-scenes dataset [46] and Cambridge landmarks [33]. Meanwhile, KITTI [35], Robotcar [38], M'alaga [47], EuRoc MAV [39], NYU [41], and TUM [40] datasets are applicable to ego-motion estimation. For scene segmentation, PASCAL VOC [42], NYU [41], Synthia [43], Cityscapes [37], KITTI [35], and ADK20 [45] datasets can be utilized.

Depth Estimation with Deep Learning

Depth estimation is fundamental in a SLAM system. Model-based SLAM methods usually take advantage of camera parallax from multiple images to estimate the depth. With the development of deep learning, data-driven methods provide an alternative to depth estimation. Depth estimation with deep learning can be divided into supervised methods and unsupervised methods.

Supervised Methods

Eigen et al. [50] designed a deep neural network to perform depth estimation with a single image. It is a supervised method where the ground-truth depth map is required for network training. The network consists of two components: one for global structure prediction and one for local prediction refinement. A scale-invariant error is defined as the cost function for learning. The real scale of depth is recovered without any post-processing. The proposed method produced good results on both NYU Depth [41] and KITTI [35] datasets.

According to the perspective geometry, the size of objects scales inversely with the depth. Ladicky et al. [53] made use

of this property to transform the image into the canonical depth for training. They also proposed to combine semantic segmentation and depth estimation together to improve the performance. The proposed method is also a supervised depth estimation method with monocular images.

Liu et al. [51] also presented a depth estimation method with single images using the so-called deep convolutional neural field (DCNF), which integrates continuous conditional random field (CRF) into a unified deep CNN framework. Further, a superpixel pooling method and fully convolutional networks (FCN) were proposed in [34] to improve the accuracy and efficiency of segmentation performance, and it could also be used for depth estimation. A similar approach was also presented by Li et al. [54].

The attempt to combine depth estimation with visual SLAM was made in [52], called CNN-SLAM. It is a monocular SLAM system in which the predicted depth map from CNN is dense and has the absolute scale. Comparing with model-based methods, only the depth is estimated from CNN while other parts, such as pose estimation and graph optimization are the same as feature-based SLAM. The proposed method demonstrated robust and accurate performance in pose estimation and map construction.

Ma et al. proposed a so-called sparse-to-dense [55] method to predict dense depth images, which could be used as a plug-in module to model-based SLAM methods to create an accurate, dense point cloud. They constructed two CNNs to fuse RGB image and sparse depth image. Their sparse depth image could be a model-based SLAM or a low-cost LiDAR.

DeMoN proposed by Ummenhofer et al. [56] also achieved depth estimation with supervised deep learning.

Supervised methods require a large amount of labeled data to train the networks. Since it is costly to collect labeled datasets, their applications are limited.

Unsupervised Methods

Recently depth estimation methods using unsupervised deep learning emerge. The main idea comes from the representation capability of autoencoders. The encoder is a CNN which predicts the depth map for the left input image, and the decoder is a wrap function which synthesizes a reconstructed left image from the right input image and the predicted depth map. The reconstructed error is used as the cost function to train the CNN [48] (see Fig. 2a).

Specifically, for the overlapped area between two stereo images, every pixel in one image can find its correspondence in the other with horizontal distance H in pixel:

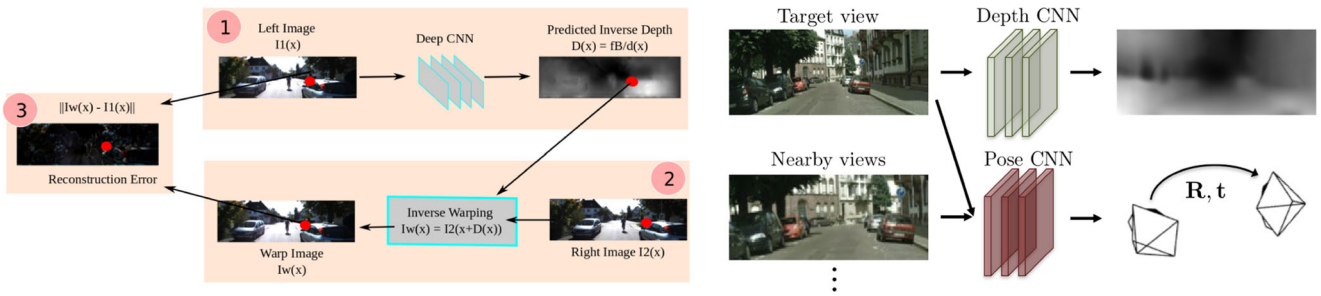
$$H = Bf/D \tag{6}$$

where B is the baseline of the stereo camera, f is the focal length, and D is the depth value of the corresponding pixel. By using the geometric constraint D map, the left image can be synthesized from the right and vice verse. Then the photometric loss function E is defined as below:

$$E = \sum \left\| I - I' \right\|_2 \tag{7}$$

where I is the original image, and $I^{'}$ is the synthesized image. By minimizing the photometric loss function E between the original left image and the synthesized left image, the network is trained fully unsupervised in an end-to-end manner. The proposed method can be viewed as a monocular depth estimation system for the reason that it only needs monocular images during testing. It even outperformed some supervised methods in terms of accuracy of depth estimation. Xie et al. [57] proposed to use a deep neural network to predict a disparity map from the left input image with unsupervised learning and then renders a novel right image for 2D-to-3D video conversion applications.

Godard et al.[14] improved the Garg's method [48] by wrapping left and right images across each other to synthesize corresponding images. In this way, the accuracy of depth prediction could be enhanced by penalizing both left and right photometric losses. Then Zhong et al. [58] presented a very similar unsupervised depth estimation system with stereo images as network inputs.



(a) Garg's method [48]. (b) SfMLearner [49].

Fig. 2 Depth estimation with unsupervised deep learning. **a** Garg's method [48]. **b** SfMLearner [49]

Zhou et al.[49] proposed SfMLearner, which uses a monocular image sequence for image alignment in order to estimate the depth and ego-motion simultaneously with unsupervised learning (see Fig. 2b). The geometric constraint between temporal image pairs is used for synthesizing corresponding images. After the training of the networks, the depth images and camera poses can be simultaneously predicted by the networks in an end-to-end manner. However, the estimated depth map and ego-motion are lack of the scale. Based on SfMLearner [49], Yang et al. [59] proposed to use a CNN to represent the surface normal map. Both predicted depth map and normal map are used to construct the loss function. Vijayanarasimhan et al.[60] presented SfM-Net which adds motion masks to the photometric loss. It can estimate optical flow, depth map, and ego-motion simultaneously.

Summary

Table 1 gives a brief summary of depth estimation methods using deep learning. The advance from supervised learning towards unsupervised learning for depth estimation is significant as no labeled data is required and it is feasible for life-long learning [48]. Unsupervised learning depth estimation is also important in building up dense maps for SLAM systems. With the view of temporal constraints in an image sequence, it is possible to estimate the ego-motion with unsupervised learning. This will be reviewed in the next section.

Pose Estimation with Deep Learning

Data-driven pose estimation with deep learning learns camera motion model and estimates poses directly without explicitly modeling. The relocalization problem is targeted first with supervised deep learning as it is possible to collect labeled data in pre-visited places. Building on the success of depth estimation with unsupervised learning, more attention has been paid to ego-motion estimation by using unsupervised learning.

Table 1 Some depth estimation methods with deep learning

Reference	Year	Feature
Eigen et al. [50]	2014	Supervised learning
Liu et al. [51]	2016	Supervised learning, CRF integrated
CNN-SLAM [52]	2017	Supervised learning, Combined with SLAM
Garg et al. [48]	2016	Unsupervised learning, stereo image, scaled depth
Godard et al. [14]	2017	Unsupervised learning, stereo image, scaled depth, left-right consistency
SfMLearner [49]	2017	Unsupervised learning, monocular image, non-scaled depth

Relocalization with Deep Learning

Most deep learning networks are originally used for classification problems. Less are found for regression problems. Kendall et al. [33] first used a CNN to solve the pose regression problem. Their PoseNet was trained with supervised learning with the requirement of ground-truth 6-DoF poses available for training (see Fig. 3a). The loss function L is designed as below:

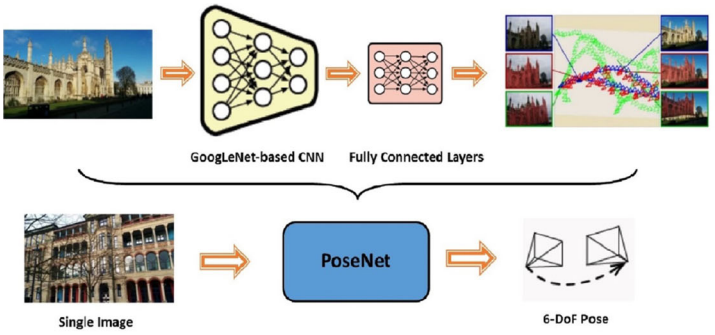
$$L = \left\| \hat{\mathbf{x}} - \mathbf{x} \right\|_2 + \lambda \left\| \hat{\mathbf{q}} - \mathbf{q} \right\|_2 \tag{8}$$

where $\mathbf{x}$ is the representation of camera position estimated by the CNN, unit quaternion $\mathbf{q}$ is the representation of camera orientation estimated by the CNN, $\hat{\mathbf{x}}$ is the ground-truth camera position, $\hat{\mathbf{q}}$ is the ground-truth camera orientation, and λ is the balance weight to normalize position and orientation losses. After supervised training with labeled data, the PoseNet could perform relocalization in pre-visited places with more robust performance than model-based methods.

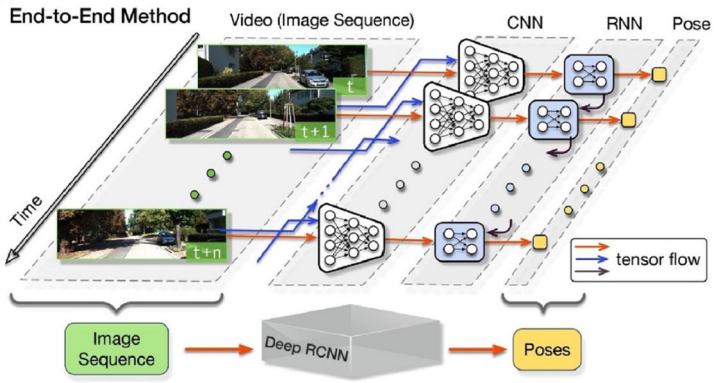
In order to estimate the uncertainty of pose estimation, Kendall et al. [64] further proposed Bayesian PoseNet by using dropout layers in the network as a means of sampling. The selection of balance weight is a trick factor in training. Kendall improved PoseNet and proposed a fusion model to train the network and automatically adjust the balance weights in [65]. The performance is improved, and the uncertainty can be estimated by the network without sampling.

Li et al. [66] then extended PoseNet from single CNN to double CNNs to accommodate color and depth inputs from RGB-D cameras. The proposed system showed robust performance when faced with challenging situations. They also applied PoseNet in night-time environment with a depth sensor [67].

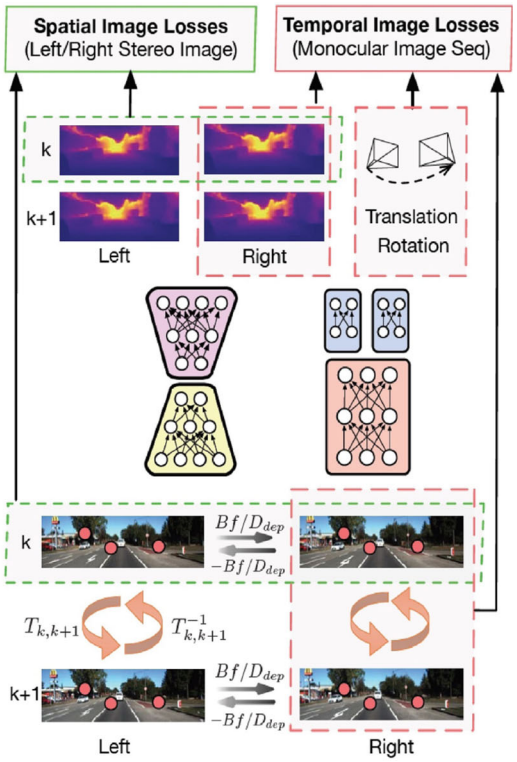
Deep RCNN architecture is adopted to explore the temporal dynamics of camera motion in pose estimation. Clark et al. [61] proposed to use an RCNN to implement the pose regression with video clips. By taking image sequences as network inputs, the uncertainty of pose estimation was reduced and relocalization performance was improved. Hazirbas et al. [68] incorporated a spatial LSTM module into PoseNet to improve the relocalization performance. The proposed system takes a single color image as input. Naseer et al. [69] adopted PoseNet [33] as the basic network and used data augmentation technology to improve the relocalization performance. They applied the transformation to synthesize additional input images. By using multiple synthesized images as inputs to perform relocalization, the system achieved better performance.



(a) PoseNet [33].



(b) DeepVO [15].



(c) UnDeepVO [62].

Fig. 3 Pose estimation with deep learning. **a** PoseNet [33]. **b** DeepVO [15]. **c** UnDeepVO [62]

Ego-Motion Estimation with Deep Learning

Apart from absolute pose regression, the ego-motion between two image frames can also be estimated by using deep learning inspired by stereo geometric models. Ego-motion estimation methods with deep learning can also be divided into supervised and unsupervised methods.

DeTone et al. proposed HomographNet [70] and used CNN to predict the homography parameters between image pairs. The proposed deep homography estimator outperformed the traditional homography estimation method based on ORB features.

Costante et al. [71] developed a CNN to estimate the ego-motion with supervised training. Wang et al. proposed monocular visual odometry system called DeepVO [15], which trains a RCNN to estimate the camera motion in an end-to-end manner. The temporal image sequence is introduced into RCNN with LSTM module (see Fig. 3b). The experiment results demonstrated a competitive performance on visual odometry. DeepVO has also been extended to incorporate uncertainty estimation [72]. Melekhov et al.[73] also presented a relative camera pose estimation system with CNN. Turan et al. proposed Deep EndoVO [74] which is similar to DeepVO [15], and applied it to the area of soft robotics [75].

Oliveira et al. [76] constructed a metric network for ego-motion estimation and a topological network for topological location estimation. The topological network discretizes the trajectory into a finite set of locations and uses the CNN to learn the topological relationship. By successfully combining this network with the ego-motion estimation network, the system demonstrated good performance in localization.

Ummenhofer et al. [56] proposed a system called DeMoN which consists of a chain of encoder-decoder networks. An iterative network is specifically designed for the system. DeMoN can estimate ego-motion, image depth, surface normal and optical flow simultaneously but needs labeled data for training.

Different from the methods using CNNs to estimate camera motions directly, DeTone et al. [63] developed two networks, one is used to estimate the location of feature points, and another one is used to match the extracted features and compute the homography. The network needs manually synthesized data for training.

Instead of predicting camera poses with a deep neural network directly, Peretroukhin et al. [77] proposed to use a model-based geometric estimator for pose prediction and a CNN for predicted pose correction. In detail, the proposed CNN is trained to learn the errors between the ground truth poses and the predicted poses from a model-based estimator.

The proposed system is called DPC-Net and can also be used for mitigating the effect of bad camera calibration parameters.

Aiming at the estimation of camera ego-motion, Costante et al. presented a novel CNN architecture which is called LS-VO [78]. LS-VO consists of an autoencoder network to learn optical flow representations, followed by a pose estimation network predicting camera poses. The networks take temporal image pairs as inputs and are trained jointly end-to-end.

In order to tackle the scale drift problem in monocular SLAM, Frost et al. [79] proposed to adopt a CNN to perform the speed regression from successive video frames. By further integrating the estimated speed into bundle adjustment, they successfully realized the scale-drift correction.

All the abovementioned methods use supervised learning schemes which require ground truth for training. Labeling large amounts of data is difficult and expensive. Therefore, it is very demanding for a visual SLAM system to learn under an unsupervised scheme so that the performance could be continuously improved by the increased size of unlabeled datasets.

Recently, [49] presented an ego-motion and depth estimation system with unsupervised deep learning. It achieves superior performance. However, the system cannot recover the absolute scale due to the use of monocular camera. Inspired by the unsupervised depth estimation methods [14, 49], Li et al. proposed UnDeepVO [62] which is a monocular visual odometry system with unsupervised learning. By using stereo pairs for training (see Fig. 3c), UnDeepVO demonstrated good performance in pose prediction and depth estimation. Further, it can also recover the absolute scale of 6-DoF poses and depth maps. Nguyen et al. [80] also introduced the similar unsupervised deep learning method into homography estimation.

Sensor Fusion with Deep Learning

Clark et al. [81] proposed a sensor fusion network called VINet, which fuses the estimated pose from DeepVO [15] and the inertial sensor reading with an LSTM. The prediction network and the fusion network are trained jointly end-to-end, and the proposed fusion system demonstrated comparable performance with traditional sensor fusion methods. Turan et al. [82, 83] adopted the same method and presented a fusion system to fuse the 6-DoF poses from cameras and magnetic sensors.

Pillai [84] proposed an ego-motion estimation system that fuses the information from a camera with other sensors such as GPS, INS, and wheel odometry. They also adopted a mixture density network to use optical flow vectors from different kinds of camera optics.

Byravan et al. [85] proposed a CNN architecture called SE3-Net, which takes raw point cloud data as input and predicts SE3 rigid transformation.

Summary

Table 2 give a brief summary of pose estimation methods using deep learning. Pose regression with CNN is a bold attempt to apply supervised deep learning for relocalization problems [33], while the ego-motion estimation [15] is a result of the capability of deep learning to capture the temporal motion dynamics. However, labeling data in large-scale hinders the application of supervised deep learning in visual SLAM systems. Unsupervised deep learning methods are powerful and promising for pose and depth estimation [49, 62].

Semantic Mapping with Deep Learning

For most autonomous robotic applications, the semantic perception of the environment is extremely important. In the computer vision community, semantic segmentation has been a long researched and established topic. When combining semantic segmentation with visual SLAM, it is possible to estimate semantic 3D map and camera motions simultaneously for robotic applications.

Semantic Segmentation

Long et al. first proposed a FCN [34] for pixel-wise semantic segmentation. The proposed system is a fully convolutional network without fully connected layer (see

Table 2 Some pose estimation methods with deep learning

Methods	Year	Reference
PoseNet [33]	2015	Relocalization, supervised learning, CNN
Clark et al. [61]	2017	Relocalization, supervised learning, RCNN
DeepVO [15]	2017	Ego-motion estimation, supervised learning, RCNN
DeMoN [56]	2017	Ego-motion estimation, supervised learning, CNN
SfMLearner [49]	2017	Ego-motion estimation, unsupervised learning, CNN, unscaled depth and pose
UnDeepVO [62]	2017	Ego-motion estimation, unsupervised learning, CNN, scaled depth and pose
DeTone et al. [63]	2017	Feature position learning

Fig. 4a). It achieved the state-of-the-art pixel-wise segmentation performance at that time. Afterward, various CNN architectures were derived from the FCN [34]. Liu et al. [93] took advantage of the global context information and introduced the global pooling into the FCN [34]. The proposed system is called ParseNet, which outperformed the FCN in scene segmentation with the wider view of the network.

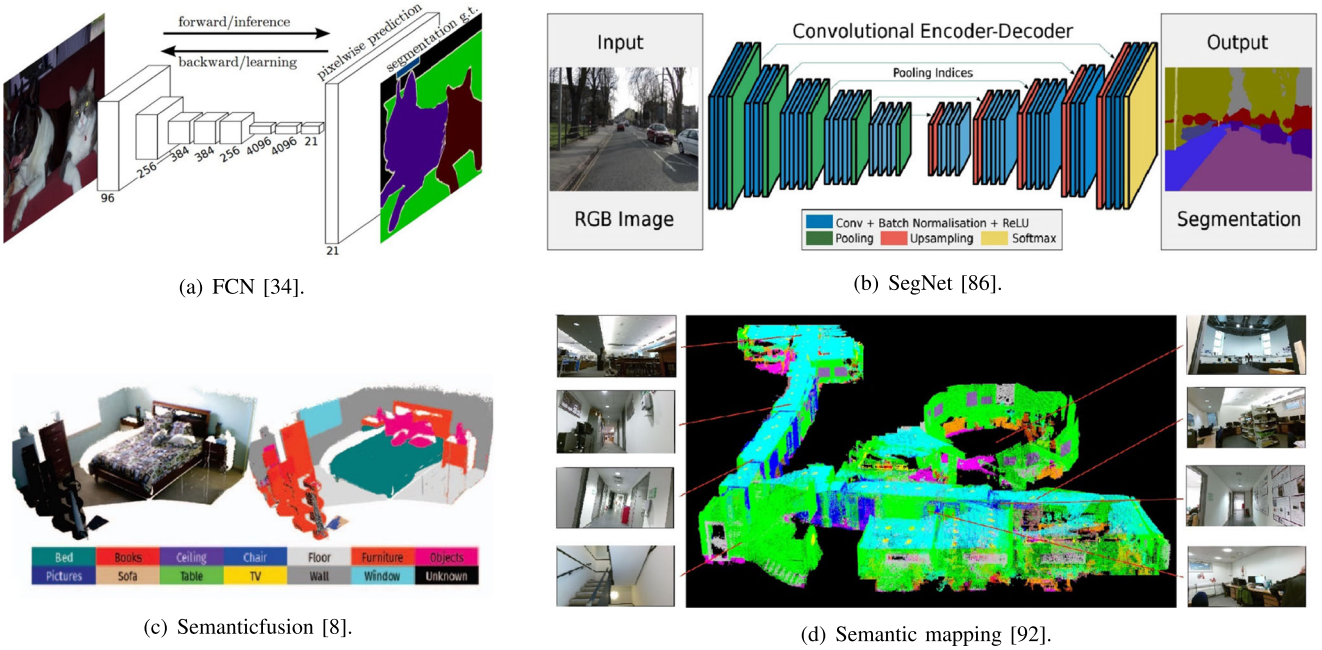
Badrinarayanan et al. [86] presented a novel network architecture called SegNet for scene segmentation. The SegNet is based on the FCN and has an encoder-decoder architecture (see Fig. 4b). The decoder performs the upsampling for low-resolution features and can recover the resolution of raw input. Afterward, Kendall et al. [94] proposed Bayesian SegNet which uses dropout layers in the SegNet as sampling. The proposed Bayesian SegNet can estimate the probability for pixel-level segmentation and achieved better performance than SegNet.

CRFs have proved its powerful capability in image segmentation and was adopted as a post-processing method to refine the image segmentation. Zheng et al. [95] proposed to formulate the probabilistic mean field inference with CRFs as RNNs. By embedding CRFs into CNNs, they presented a novel network architecture called CRF-RNN, which combines the strength of both CNNs and CRFs. Afterward, Arnab et al. [96] designed two high-order potentials based on object detection and superpixels, and integrated them into the CRF-RNN. However, CRFs are especially computational intensive and not suitable for real-time applications.

The networks mentioned above all used the VGG [97] as their base network architecture. After He et al. proposed a very deep ResNet [98], it has gradually become the basic network architecture. The ResNet demonstrated an astonishing performance in the ImageNet classification challenge [99] and has been widely applied for many semantic segmentation tasks. Chen et al. [87, 100, 101] proposed to use the very deep ResNet, dilated convolution, and fully connected CRFs to perform image segmentation. By using dilated convolution [102], the field-of-view of filters could be enlarged effectively without increasing the computation. Atrous spatial pyramid pooling (ASPP) and multiple scale technologies were also introduced in Deeplab [87], which performed extremely well in PASCAL VOC-2012 [103] semantic image segmentation dataset.

Wu et al. [104] explored variations of the ResNet in order to find the best network configuration, such as the number of layers, the size of field-of-view and the resolution of feature maps. An online bootstrapping method is also used during training to improve the segmentation performance. The proposed network was evaluated on both PASCAL VOC-2012 benchmark and Cityscapes [37] benchmark. The results show that the proposed network is very competitive when compared with other methods.

Afterward, Wu et al. [88] further studied the relationship between the depth of residual networks and the performance, and proved that some relatively shallow residual networks could outperform much deeper networks, particularly within some limitations. This performance is not only



(a) FCN [34].

(b) SegNet [86].

(c) Semanticfusion [8].

(d) Semantic mapping [92].

Fig. 4 Semantic mapping with deep learning. **a** FCN [34]. **b** SegNet [86]. **c** Semanticfusion [8]. **d** Semantic mapping [92]

applied to the recognition task but also suitable for the semantic segmentation task.

Zhao et al. [89] proposed the pyramid scene parsing network (PSPNet) which won the ImageNet scene parsing challenge 2016 [45]. Different from the global pooling method proposed in [93], the global spatial context information in images was exploited by different-region-based aggregation with the proposed pyramid pooling model in [89]. Liu et al. [105] also used deep fully convolutional residual network with pyramid pooling for road segmentation.

Based on the SegNet [86], Hazirbas et al. [106] extracted features from color images and depth images, respectively, and fused them together to perform upsampling. Both color features and depth features are exploited for segmentation with this FuseNet [106]. AdapNet was proposed by Valada et al. [107, 108] for semantic segmentation in adverse environments. A novel fusion technology called Convoluted Mixture of Deep Experts (CMoDE) was presented to enable a multi-stream network to learn features from different modalities.

Semantic Mapping

Semantic information is particularly valued in robot-human and robot-environment interaction [109]. With the progress in semantic segmentation using deep learning, semantic SLAM research grows rapidly. Li et al. [92] combined model-based SLAM methods with spatio-temporal CNN-based semantic segmentation (see Fig. 4d). The proposed system can perform 3D semantic scene mapping and 6-DoF localization simultaneously. The system could perform in large indoor environments. A similar semantic mapping system with the pixel-voxel network was proposed by Zhao et al. [91]. McCormac et al. [8] proposed SemanticFusion which integrates CNN-based semantic segmentation with the dense SLAM technology ElasticFusion. SemanticFusion can perform in indoor scenes and produce a dense 3D semantic map (see Fig. 4c).

Summary

Table 3 gives a brief summary of semantic mapping methods using deep learning. Semantic SLAM is very challenging without the use of deep learning. The success in semantic segmentation, such as FCN [34] SegNet [86] Deeplab [88, 95], boosts the research in semantic SLAM [8, 91, 92]. It is expected more fruitful results will be generated in coming years. However, the most significant challenge in semantic SLAM is the supervised learning which requires a large amount of labeled dataset to train the networks. Weekly supervised learning for semantic SLAM may be made available due to the achieved breakthroughs in weekly supervised semantic segmentation [110].

Table 3 Some semantic perception methods with deep learning

Methods	Year	Reference
FCN [34]	2015	Semantic segmentation, VGG-based
SegNet [86]	2015	Semantic segmentation, VGG-based, encoder-decoder architecture
Deeplab [87]	2016	Semantic segmentation, ResNet-based, dilated convolution, CRF
Wu et al. [88]	2016	Semantic segmentation, ResNet-based, online bootstrapping
PSPNet [89]	2016	Semantic segmentation, ResNet-based, dilated convolution, pyramid pooling model
Semanticfusion [8]	2017	Semantic mapping, ElasticFusion-based
Li et al. [90]	2017	Semantic mapping, ORB-SLAM-based, spatio-temporal CNN for segmentation
Zhao et al. [91]	2017	Semantic mapping, RGB-D SLAM-based,pixel-voxel CNN for segmentation

Open Challenges and Future Opportunities

Given the success of model-based SLAM methods in the high accuracy of localization and mapping, the improvement of robustness, the integration of semantic information, and the incorporation of learning capability become the core of next step development in visual SLAM systems. Deep learning-based methods are demonstrating the potential in all of these aspects. We visualize the following opportunities for future development along this trend.

ImageNet-Scale Dataset for Learning-Based Visual SLAM

Most of deep learning-based methods are based on supervised learning schemes which require labeled datasets. However, labeling a large amount of data is time-consuming and labor-intensive, which limits the potential application scenarios of deep learning-based methods. This is particularly true in the context of visual SLAM

because robots or autonomous systems typically operate in completely unknown environments.

Current results show the robustness of supervised deep learning-based methods are able to outperform model-based ones in some challenging scenes. However, large-scale labeled datasets are the bottleneck for further development. It is appealing yet hard to get ImageNet-scale dataset for all visual SLAM applications.

Therefore, it is very demanding for a visual SLAM system to learn under an unsupervised scheme. The performance could be continuously improved by the increased size of unlabeled datasets. As unsupervised deep learning methods [49, 62] has already shown some promising results, it will be very interesting to see how the performance of visual SLAM changes as the size of the training dataset increases. Unsupervised deep learning is expected to exploit a truly large-scale data, boosting the capability of visual SLAM in terms of robustness, and semantic understanding.

Semantic SLAM with High-Level Understanding

For intelligent robots or autonomous systems, understanding semantic information is essential and important. FCNs have produced the state-of-the-art results on pixel-wise semantic segmentation in the last few years.

Object-level semantic SLAM methods with deep learning will play a significant role in large-scale and complex environments. Objects can be extracted from the geometric 3D map produced from visual SLAM systems. Further understanding object properties and mutual relations will enable a better interaction between robots and human or robots and environments. Moreover, object-level semantic information has the potential to improve the accuracy and robustness of pose estimation while pose estimation can do the same for semantic segmentation [8, 91, 92].

With the aid of high-level understanding of the scenarios, task-driven SLAM which could provide high efficiency and wide generalization is also a promising area to explore.

Adaptive SLAM Methods for Different Sensing Modalities

Different kinds of sensors bring in different features of the environments. How to take the most advantage of each of them in visual SLAM has always been a big question to answer. Apart from conventional optimal state estimation based multi-sensor fusion, sensor fusion and management in the framework of deep learning is being proved increasingly useful. Learning-based methods potentially generate new adaptive visual SLAM paradigms which can accommodate different sensing modalities to replace the calibration process.

Integration of Model-Based Methods with Deep Learning

Model-based SLAM methods have already achieved great success. However, they heavily depend on the successful detection of features. Most existing features, such as SIFT, SURF, or ORB features, are still fragile when encountering featureless or challenging scenes. The powerful representation capability in deep learning can be used to extract more robust scale-invariant, lighting-invariant, and rotation-invariant features. Extracting and matching features robustly by using supervised deep learning have already been reported in [63]. Exploring the use of unsupervised learning for extracting and matching more robust features is a means to improve model-based methods.

Maintaining a globally consistent map is a very important component of any SLAM system. For model-based methods, graph-based pose optimization and global bundle adjustment are the keys to gain the high accuracy by maintaining a consistent map in the back-end. How to use deep learning methods to maintain a globally consistent map is still an open question to answer.

Conclusions

The maturity of model-based SLAM in accuracy leads to seeking the robustness and the high-level cognition and perception within visual SLAM systems. The attention is being gradually turned towards a deep learning solution inspired by the powerful capability of deep learning in various visual tasks. Additionally, a visual SLAM system with learning or adaptive capability is an attractive factor for further exploration. Moreover, the deep learning solution can also make a visual SLAM system more flexible in producing a variety of meaningful estimated results, such as pose, depth, 3D point cloud, and semantic map.

We have provided significant evidence to show the ongoing evolution from model-based to deep learning-based methods is happening. The performance in improving robustness, integrating semantic information, and incorporating leaning capability has been demonstrated in some of deep learning solutions. It is expected more fruitful results will continue to come.

The knowledge of model-based visual SLAM is valued in designing the network architecture, the loss function, and the data representation of deep learning-based methods. The availability of large-scale dataset is a key to broad applications of deep learning methods. The attempt to employ unsupervised learning is promising to further consolidate the deep learning contribution to visual SLAM.

Acknowledgments The authors are grateful to the reviewers for their valuable comments that considerably contributed to improving this paper.

Funding Information The first author has been financially supported by scholarship from China Scholarship Council.

Compliance with Ethical Standards

Conflict of interests The authors declare that they have no conflict of interest.

Informed Consent Informed consent was obtained from all individual participants included in the study.

Human and Animal Rights This article does not contain any studies with human or animal subjects performed by the any of the authors.

References

1. Klein G, Murray D. Parallel tracking and mapping for small AR workspaces. In: IEEE/ACM International Symposium on Mixed and Augmented Reality. IEEE; 2007. p. 225–234.
2. Endres F, Hess J, Sturm J, Cremers D, Burgard W. 3-D Mapping with an RGB-d camera. IEEE Trans Robot. 2014;30(1):177–187.
3. Mur-Artal R, Montiel J, Tardos JD. ORB-SLAM: a versatile and accurate monocular SLAM system. IEEE Trans Robot. 2015;31(5):1147–1163.
4. Newcombe RA, Lovegrove SJ, Davison AJ. DTAM: dense tracking and mapping in real-time. In: IEEE International Conference on Computer Vision (ICCV). IEEE; 2011. p. 2320–2327.
5. Forster C, Pizzoli M, Scaramuzza D. SVO: fast semi-direct monocular visual odometry. In: IEEE International Conference on Robotics and Automation (ICRA). IEEE; 2014. p. 15–22.
6. Engel J, Koltun V, Cremers D. Direct sparse odometry. IEEE Trans Patt Anal Mach Intell. 2018;40(3):611–25.
7. Cadena C, Carlone L, Carrillo H, Latif Y, Scaramuzza D, Neira J, Reid I, Leonard JJ. Past, present, and future of simultaneous localization and mapping: toward the robust-perception age. IEEE Trans Robot. 2016;32(6):1309–1332.
8. McCormac J., Handa A., Davison A., Leutenegger S. SemanticFusion: dense 3D semantic mapping with convolutional neural networks. In: IEEE International Conference on Robotics and Automation (ICRA). IEEE; 2017. p. 4628–4635.
9. Goodfellow I, Bengio Y, Courville A. Deep learning. Cambridge: MIT Press; 2016.
10. LeCun Y, Boser B, Denker JS, Henderson D, Howard RE, Hubbard W, Jackel LD. Backpropagation applied to handwritten zip code recognition. Neural Comput. 1989;1(4):541–551.
11. Perera AG, Law YW, Chahl J. Human pose and path estimation from aerial video using dynamic classifier selection, Cognitive Computation. 2018. https://doi.org/10.1007/s12559-018-9577-6.
12. Cao L, Sun F, Liu X, Huang W, Kotagiri R, Li H. End-to-end convnet for tactile recognition using residual orthogonal tiling and pyramid convolution ensemble, Cognitive Computation. 2018. https://doi.org/10.1007/s12559-018-9568-7.
13. Zeng D, Zhao F, Shen W, Ge S. Compressing and accelerating neural network for facial point localization. Cogn Comput. 2018;10(2):359–367.
14. Godard C, Mac Aodha O, Brostow GJ. Unsupervised monocular depth estimation with left-right consistency. In: IEEE Conference on computer vision and pattern recognition (CVPR); 2017.
15. Wang S, Clark R, Wen H, Trigoni N. DeepVO: towards end-to-end visual odometry with deep recurrent convolutional neural networks. In: IEEE International Conference on Robotics and Automation (ICRA). IEEE; 2017. p. 2043–2050.
16. Durrant-Whyte H, Bailey T. Simultaneous localization and mapping: Part I. IEEE Robot Autom Magazine. 2006;13(2):99–110.
17. Bailey T, Durrant-Whyte H. Simultaneous localization and mapping: part II. IEEE Robot Autom Magazine. 2006;13(3):108–117.
18. Scaramuzza D, Fraundorfer F. Visual odometry: part I - the first 30 years and fundamentals. IEEE Robot Autom Magazine. 2011;18(4):80–92.
19. Fraundorfer F, Scaramuzza D. Visual odometry: part II - matching, robustness, optimization, and applications. IEEE Robot Autom Magazine. 2012;19(2):78–90.
20. Davison AJ, Reid ID, Molton ND, Stasse O. MonoSLAM: real-time single camera SLAM. IEEE Trans Patt Anal Mach Intell. 2007;29(6):1052–1067.
21. Mur-Artal R, Tardós JD. Fast relocalisation and loop closing in keyframe-based SLAM. In: IEEE International Conference on Robotics and Automation (ICRA). IEEE; 2014. p. 846–853.
22. Rublee E, Rabaud V, Konolige K, Bradski G. ORB: an efficient alternative to SIFT or SURF. In: IEEE international conference on Computer Vision (ICCV). IEEE; 2011. p. 2564–2571.
23. Mur-Artal R, Tardós JD. ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-d cameras. IEEE Trans Robot. 2017;33(5):1255–1262.
24. Kueng B, Mueggler E, Gallego G., Scaramuzza D. Low-latency visual odometry using event-based feature tracks. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE; 2016. p. 16–23.
25. Kim H, Leutenegger S, Davison AJ. Real-time 3D reconstruction and 6-DoF tracking with an event camera. In: European Conference on Computer Vision. Springer; 2016. p. 349–364.
26. Salas-Moreno RF, Glocken B, Kelly PH, Davison AJ. Dense planar SLAM. In: IEEE International Symposium on Mixed and Augmented Reality (ISMAR). IEEE; 2014. p. 157–164.
27. Salas-Moreno RF, Newcombe RA, Strasdat H, Kelly PH, Davison AJ. Slam++: simultaneous localisation and mapping at the level of objects. In: IEEE Conference on Computer Vision and Pattern Recognition; 2013. p. 1352–1359.
28. Newcombe RA, Izadi S, Hilliges O, Molyneaux D, Kim D, Davison AJ, Kohi P, Shotton J, Hodges S, Fitzgibbon A. KinectFusion: real-time dense surface mapping and tracking. In: IEEE international symposium on Mixed and Augmented Reality (ISMAR). IEEE; 2011. p. 127–136.
29. Whelan T, Salas-Moreno RF, Glocker B, Davison AJ, Leutenegger S. ElasticFusion: real-time dense SLAM and light source estimation. Int J Robot Res. 2016;35(14):1697–1716.
30. Engel J, Sturm J, Cremers D. Semi-dense visual odometry for a monocular camera. In: IEEE International Conference on Computer Vision; 2013. p. 1449–1456.
31. Engel J, Schöps T, Cremers D. LSD-SLAM: large-scale direct monocular SLAM. In: European Conference on Computer Vision (ECCV). Springer; 2014. p. 834–849.
32. Pascoe G, Maddern W, Tanner M, Piniés P, Newman P. NID-SLAM: robust monocular SLAM using normalised information distance. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2017.

33. Kendall A, Grimes M, Cipolla R. PoseNet: a convolutional network for real-time 6-DoF camera relocalization. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV); 2015. p. 2938–2946.
34. Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition; 2015. p. 3431–3440.
35. Geiger A, Lenz P, Urtasun R. Are we ready for autonomous driving? The KITTI vision benchmark suite. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2012.
36. Janai J, Güney F, Behl A, Geiger A. Computer vision for autonomous vehicles: problems, datasets and state-of-the-art. 2017. arXiv:1704.05519.
37. Cordts M, Omran M, Ramos S, Rehfeld T, Enzweiler M, Benenson R, Franke U, Roth S, Schiele B. The cityscapes dataset for semantic urban scene understanding. In: IEEE Conference on Computer Vision and Pattern Recognition; 2016. p. 3213–3223.
38. Maddern W, Pascoe G, Linegar C, Newman P. 1 Year, 1000km: the Oxford robotCar dataset. The International Journal of Robotics Research (IJRR). 2017;36(1):3–15.
39. Burri M, Nikolic J, Gohl P, Schneider T, Rehder J, Omari S, Achtelik MW, Siegwart R. The EuRoC micro aerial vehicle datasets. Int J Robot Res. 2016;35(10):1157–1163.
40. Sturm J, Engelhard N, Endres F, Burgard W, Cremers D. A benchmark for the evaluation of RGB-D SLAM systems. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE; 2012. p. 573–580.
41. Silberman N, Hoiem D, Kohli P, Fergus R. Indoor segmentation and support inference from RGBD images. In: European Conference on Computer Vision; 2012. p. 746–760.
42. Everingham M, Van Gool L, Williams CK, Winn J, Zisserman A. The pascal visual object classes (VOC) challenge. Int J Comput Vis. 2010;88(2):303–338.
43. Ros G, Sellart L, Materzynska J, Vazquez D, Lopez AM. The synthia dataset: a large collection of synthetic images for semantic segmentation of urban scenes. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2016. p. 3234–3243.
44. Lin T-Y, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Dollár P, Zitnick CL. Microsoft COCO: common objects in context. In: European Conference on Computer Vision. Springer; 2014. p. 740–755.
45. Zhou B, Zhao H, Puig X, Fidler S, Barriuso A, Torralba A. Semantic understanding of scenes through the ADE20K dataset. 2016. arXiv:1608.05442.
46. Shotton J, Glocker B, Zach C, Izadi S, Criminisi A, Fitzgibbon A. Scene coordinate regression forests for camera relocalization in RGB-D images. In: IEEE Conference on Computer Vision and Pattern Recognition; 2013. p. 2930–2937.
47. Blanco-Claraco J-L, Moreno-Dueñas F-Á, González-Jiménez J. The Málaga urban dataset: high-rate stereo and liDAR in a realistic urban scenario. Int J Robot Res. 2014;33(2):207–214.
48. Garg R, Carneiro G, Reid I. Unsupervised CNN for single view depth estimation: geometry to the rescue. In: European Conference on Computer Vision (ECCV). Springer; 2016. p. 740–756.
49. Zhou T, Brown M, Snavely N, Lowe DG. Unsupervised learning of depth and ego-motion from video. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2017.
50. Eigen D, Puhrsch C, Fergus R. Depth map prediction from a single image using a multi-scale deep network. In: Advances in Neural Information Processing Systems; 2014. p. 2366–2374.
51. Liu F, Shen C, Lin G, Reid I. Learning depth from single monocular images using deep convolutional neural fields. IEEE Trans Pattern Anal Mach Intell. 2016;38(10):2024–2039.
52. Tateno K, Tombari F, Laina I, Navab N. CNN-SLAM: real-time dense monocular SLAM with learned depth prediction. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2017.
53. Ladicky L, Shi J, Pollefeys M. Pulling things out of perspective. In: IEEE Conference on Computer Vision and Pattern Recognition; 2014. p. 89–96.
54. Li B, Shen C, Dai Y, van den Hengel A, He M. Depth and surface normal estimation from monocular images using regression on deep features and hierarchical CRFs. In: IEEE Conference on Computer Vision and Pattern Recognition; 2015. p. 1119–1127.
55. Ma F, Karaman S. Sparse-to-dense: depth prediction from sparse depth samples and a single image. 2017. arXiv:1709.07492.
56. Ummenhofer B, Zhou H, Uhrig J, Mayer N, Ilg E, Dosovitskiy A, Brox T. Demon: depth and motion network for learning monocular stereo. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2017.
57. Xie J, Girshick R, Farhadi A. Deep3d: fully automatic 2D-to-3D video conversion with deep convolutional neural networks. In: European Conference on Computer Vision (ECCV). Springer; 2016. p. 842–857.
58. Zhong Y, Dai Y, Li H. Self-supervised learning for stereo matching with self-improving ability. 2017. arXiv:1709.00930.
59. Yang Z, Wang P, Xu W, Zhao L, Nevatia R. Unsupervised learning of geometry with edge-aware depth-normal consistency. 2017. arXiv:1711.03665.
60. Vijayanarasimhan S, Ricco S, Schmid C, Sukthankar R, Fragkiadaki K. Sfm-net: learning of structure and motion from video. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2017.
61. Clark R, Wang S, Markham A, Trigoni N, Wen H. Vidloc: 6-DoF video-clip relocalization. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2017.
62. Li R, Wang S, Long Z, Gu D. Undeepvo: monocular visual odometry through unsupervised deep learning. 2017. arXiv:1709.06841.
63. DeTone D, Malisiewicz T, Rabinovich A. Toward geometric deep SLAM. 2017. arXiv:1707.07410.
64. Kendall A, Cipolla R. Modelling uncertainty in deep learning for camera relocalization. In: IEEE International Conference on Robotics and Automation (ICRA). IEEE; 2016. p. 4762–4769.
65. Kendall A, Cipolla R. Geometric loss functions for camera pose regression with deep learning. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2017.
66. Li R, Liu Q, Gui J, Gu D, Hu H. Indoor relocalization in challenging environments with dual-stream convolutional neural networks. IEEE Trans Autom Sci Eng. 2018;15(2):651–62.
67. Li R, Liu Q, Gui J, Gu D, Hu H. Night-time indoor relocalization using depth image with convolutional neural networks. In: International Conference on Automation and Computing (ICAC). IEEE; 2016. p. 261–266.
68. Hazirbas FWC, Sattler LL-TT, Hilsenbeck S, Cremers D. Image-based localization using LSTMs for structured feature correlation.
69. Naseer T, Burgard W. Deep regression for monocular camera-based 6-DoF global localization in outdoor environments.
70. DeTone D, Malisiewicz T, Rabinovich A. Deep image homography estimation. 2016. arXiv:1606.03798.
71. Costante G, Mancini M, Valigi P, Ciarfuglia TA. Exploring representation learning with CNNs for frame-to-frame ego-motion estimation. IEEE Robot Autom Lett. 2016;1(1):18–25.

72. Wang S, Clark R, Wen H, Trigoni N. End-to-end, sequence-to-sequence probabilistic visual odometry through deep neural networks. Int J Robot Res. 2018;37(4-5):513–42.
73. Melekhov I, Kannala J, Rahtu E. Relative camera pose estimation using convolutional neural networks. 2017. arXiv:1702.01381.
74. Turan M, Almalioglu Y, Araujo H, Konukoglu E, Sitti M. Deep endovo: A recurrent convolutional neural network (rcnn) based visual odometry approach for endoscopic capsule robots. Neurocomputing. 2018;275:1861–70.
75. Zhao H, O'Brien K, Li S, Shepherd RF. Optoelectronically innervated soft prosthetic hand via stretchable optical waveguides. Sci Robot. 2016;1(1):eaai7529.
76. Oliveira GL, Radwan N, Burgard W, Brox T. Topometric localization with deep learning. 2017. arXiv:1706.08775.
77. Peretroukhin V, Kelly J. DPC-Net: Deep pose correction for visual localization. 2017. arXiv:1709.03128.
78. Costante G, Ciarfuglia TA. LS-VO: Learning dense optical subspace for robust visual odometry estimation. In: IEEE Robotics and Automation Letters, 2018; Vol. 3, no. 3, p. 1735–1742. https://doi.org/10.1109/LRA.2018.2803211.
79. Frost DP, Murray DW, Prisacariu VA. Using learning of speed to stabilize scale in monocular localization and mapping.
80. Nguyen T, Chen SW, Shivakumar SS, Taylor CJ, Kumar V. Unsupervised deep homography: a fast and robust homography estimation model. 2017. arXiv:1709.03966.
81. Clark R, Wang S, Wen H, Markham A, Trigoni N. VINet: visual-inertial odometry as a sequence-to-sequence learning problem. In: AAAI; 2017. p. 3995–4001.
82. Turan M, Almalioglu Y, Gilbert H, Sari AE, Soylu U, Sitti M. Endo-VMFuseNet: Deep visual-magnetic sensor fusion approach for uncalibrated, unsynchronized and asymmetric endoscopic capsule robot localization data. 2017. arXiv:1709.06041.
83. Turan M, Almalioglu Y, Araujo H, Cemgil T, Sitti M. Endosensorfusion: particle filtering-based multi-sensory data fusion with switching state-space model for endoscopic capsule robots using recurrent neural network kinematics. 2017. arXiv:1709.03401.
84. Pillai S, Leonard JJ. Towards visual ego-motion learning in robots. 2017. arXiv:1705.10279.
85. Byravan A, Fox D. SE3-Nets: learning rigid body motion using deep neural networks. In: IEEE International Conference on Robotics and Automation (ICRA). IEEE; 2017. p. 173–180.
86. Badrinarayanan V, Kendall A, Cipolla R. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE Trans Pattern Anal Mach Intell. 2017;39(12):2481–95.
87. Chen L-C, Papandreou G, Kokkinos I, Murphy K, Yuille AL. Deeplab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. 2016. arXiv:1606.00915.
88. Wu Z, Shen C, Hengel Avd. Wider or deeper: revisiting the resnet model for visual recognition. 2016. arXiv:1611.10080.
89. Zhao H, Shi J, Qi X, Wang X, Jia J. Pyramid scene parsing network. 2016. arXiv:1612.01105.
90. Li R, Gu D, Liu Q, Long Z, Hu H. Semantic scene mapping with spatio-temporal deep neural network for robotic applications, Cognitive Computation. 2017. https://doi.org/10.1007/s12559-017-9526-9.
91. Zhao C, Sun L, Shuai B, Purkait P, Stolkin R. Dense RGB-D semantic mapping with pixel-voxel neural network. 2017. arXiv:1710.00132.
92. Li R, Gu D, Liu Q, Long Z, Hu H. Semantic scene mapping with spatio-temporal deep neural network for robotic applications. Cogn Comput. 2018;10(2):260–271.
93. Liu W, Rabinovich A, Berg AC. ParseNet: looking wider to see better. 2015. arXiv:1506.04579.
94. Kendall A, Badrinarayanan V, Cipolla R. Bayesian segnet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding. 2015. arXiv:1511.02680.
95. Zheng S, Jayasumana S, Romera-Paredes B, Vineet V, Su Z, Du D, Huang C, Torr PH. Conditional random fields as recurrent neural networks. In: IEEE International Conference on Computer Vision; 2015. p. 1529–1537.
96. Arnab A, Jayasumana S, Zheng S, Torr PH. Higher order conditional random fields in deep neural networks. In: European Conference on Computer Vision. Springer; 2016. p. 524–540.
97. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. In: International Conference on Learning Representations; 2015. p. 1–14.
98. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition; 2016. p. 770–778.
99. Deng J, Dong W, Socher R, Li L-J, Li K, Fei-Fei L. Imagenet: a large-scale hierarchical image database. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE; 2009. p. 248–255.
100. Chen L-C, Papandreou G, Kokkinos I, Murphy K, Yuille AL. Semantic image segmentation with deep convolutional nets and fully connected CRFs. 2014. arXiv:1412.7062.
101. Chen L-C, Yang Y, Wang J, Xu W, Yuille AL. Attention to scale: scale-aware semantic image segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition; 2016. p. 3640–3649.
102. Yu F, Koltun V. Multi-scale context aggregation by dilated convolutions. 2015. arXiv:1511.07122.
103. Everingham M, Eslami SA, Van Gool L, Williams CK, Winn J, Zisserman A. The pascal visual object classes challenge: a retrospective. Int J Comput Vis. 2015;111(1):98–136.
104. Wu Z, Shen C, Hengel Avd. High-performance semantic segmentation using very deep fully convolutional networks. 2016. arXiv:1604.04339.
105. Liu X, Deng Z. Segmentation of drivable road using deep fully convolutional residual network with pyramid pooling. Cogn Comput. 2018;10(2):272–281.
106. Hazirbas C, Ma L, Domokos C, Cremers D. Fusenet: Incorporating depth into semantic segmentation via fusion-based CNN architecture. In: Asian conference on computer vision; 2016.
107. Valada A, Oliveira G, Brox T, Burgard W. Towards robust semantic segmentation using deep fusion. In: Robotics: Science and systems (RSS 2016) Workshop, Are the Sceptics Right? Limits and Potentials of Deep Learning in Robotics; 2016.
108. Valada A, Vertens J, Dhall A, Burgard W. Adapnet: adaptive semantic segmentation in adverse environmental conditions. In: IEEE International conference on robotics and automation (ICRA). IEEE; 2017.
109. Hülse M., McBride S, Lee M. Fast learning mapping schemes for robotic hand–eye coordination. Cogn Comput. 2010;2(1):1–16.
110. Pathak D, Krahenbuhl P, Darrell T. Constrained convolutional neural networks for weakly supervised segmentation. In: Proceedings of the IEEE international conference on computer vision; 2015. p. 1796–1804.