Transportation Safety and Environment, 2019, Vol. 1, No. 3 177–184

doi: 10.1093/tse/tdz019
Review

REVIEW

Deep Learning for Visual SLAM in Transportation Robotics: A review

Chao Duan[1], Steffen Junginger[2], Jiahao Huang[1], Kairong Jin[1] and Kerstin Thurow[3,*]

[1]Institute of Artificial Intelligence & Robotics (IAIR), School of Traffic & Transportation Engineering, Central South University, 410075 China [2]Institute of Automation, University Rostock, 18119 Germany [3]Center for Life Science Automation, University of Rostock, 18119 Germany

*Corresponding author. E-mail: Kerstin.Thurow@celisca.de

Abstract

Visual SLAM (Simultaneously Localization and Mapping) is a solution to achieve localization and mapping of robots simultaneously. Significant achievements have been made during the past decades, geography-based methods are becoming more and more successful in dealing with static environments. However, they still cannot handle a challenging environment. With the great achievements of deep learning methods in the field of computer vision, there is a trend of applying deep learning methods to visual SLAM. In this paper, the latest research progress of deep learning applied to the field of visual SLAM is reviewed. The outstanding research results of deep learning visual odometry and deep learning loop closure detect are summarized. Finally, future development directions of visual SLAM based on deep learning is prospected.

Keywords: deep learning; visual SLAM; transportation robotics; mobile robots

1. Introduction

Mobile robot autonomous positioning and navigation needs to solve three problems: localization, mapping and path planning [1]. Localization includes the robot pose and location in the environment. Mapping is used for generating a representation of the surrounding environment, while path planning solves the problem of the robot moving in the optimal route in the map. In the early days, localization and mapping were studied separately until the IEEE Robotics and

Received: November 1, 2019. **Revised:** November 29, 2019. **Accepted:** December 7, 2019

Automation Conference in 1986 proposed the concept of SLAM (Simultaneously Localization and Mapping). SLAM can solve the problem that the robot locates itself in an unknown environment and gradually construct a continuous map of the environment. In case of a camera as the only external sensor, this concept is called visual SLAM.

State-of-the-art visual SLAM frameworks consist of the following four modules [2]: frontend visual odometry, backend optimization, loop closure detection, and mapping. Visual odometry is responsible for preliminarily estimating the pose and location of the robot frame to frame and the position of the map point. The backend optimization is responsible for receiving the pose information measured by visual odometry and calculating Maximum a posteriori estimation (MAP). The loop closure detection aims at recognizing the previously visited places during the travel of the mobile robot, and help the mapping and registration algorithms to obtain a more accurate and consistent result. Finally, the mapping is responsible for reconstructing the map according to the camera pose and frame.

The main applications of visual SLAM are around robotics, including autonomous vehicles [3], Unmanned Aerial Vehicles (UVAs) [4], underwater robots [5], medicine [6], and augmented reality [7].

Many visual SLAM systems fail while working in external environments, in dynamic environments, in environments with too many or very few salient features, in large scale environments, during erratic movements of the camera and when partial or if a total occlusions of the sensor occurs.

As it is known, deep learning has promoted breakthroughs in research on image recognition [8] and speech recognition [9], opening the era of "big data + complex models". The success of deep learning should be attributed to deep model structure, efficient learning methods, support for big data, and ever-changing computing power.

Unavoidably, the evolution of visual SLAM from geometry-based methods to deep learning methods occurs. Recently both supervised deep learning methods and unsupervised methods are applied for visual SLAM problems such as visual odometry [10, 11] and loop closure [12, 13]. These recent advances promise huge potential for deep learning methods to address the challenging issues of visual SLAM by including adaptive and learning capability.

This paper provides a review of deep learning methods applied on visual SLAM, including the advantagse and limitations of the different deep learning methods. Finally, future opportunities are focusing on the way to enhance robustness, semantic understanding, and learning capability of visual SLAM.

The paper is organized as follows. In chapter 2 we review the related work of visual SLAM method based on geometry briefly, chapter 3 illustrates deep learning VO methods. Chapter 4 illustrates loop closure detection methods based on deep learning methods. Then the open problem and development trend of visual SLAM is discussed in chapter 5. Finally, a conclusion is drawn in chapter 6.

2. Visual SLAM based on Geometry

In theory, the visual SLAM method based on geometric theory mainly relies on extracting geometric constraints from images to estimate motions. Since they come from elegant and established principles and are widely investigated, most of the state-of-the-art visual SLAM algorithms belong to this family. They can be further divided into feature-based methods and direct methods [14].

2.1 Feature-based Method

First feature-based monocular visual SLAM was presented in 2003 by Davison et al., which is called MonoSLAM [15]. MonoSLAM is considered to be a typical filter-based visual SLAM method. In MonoSLAM, an extended Kalman filter (EKF) is used to simultaneously estimate the camera motion and the 3D structure of an unknown environment. The 6-degree-of-freedom (DoF) motion of the camera and the 3D position of the feature point are represented as a state vector in the EKF. In the prediction model, it is assumed that the motion is uniform, and the result of the feature point tracking is taken as the observation result. New feature points are added to the state vector based on camera movement. The problem with this approach is that the amount of computation increases with the size of the environment. In a large environment, the size of the state vector increases due to the large number of feature points. In this case, it is difficult to achieve real-time calculations.

To solve the computational problem in MonoSLAM, PTAM [2] splits the tracking and mapping into different threads on CPU. These two threads are executed in parallel. Thus, the

computational cost of the mapping does not affect the tracking. Therefore, a BA (Bundle Adjustment) [16] that requires a computational amount in the optimization can be used for mapping. This enables tracking the motion of the camera in real-time, and mapping the estimated 3D position of the feature points with the amount of computation. PTAM is the first method to integrate BA into a real-time visual SLAM algorithm. One of the important contributions of PTAM is the introduction of keyframe-based mapping in visual SLAM. After the release of PTAM, most visual SLAM algorithms follow this type of multi-threading method including ORB-SLAM and LSD-SLAM.

ORB-SLAM proposed by Mur-Artal et al. [17] inherits the framework of PTAM and replaces most of the modules, which is one of the most successful feature-based visual SLAM systems by now. Fig. 1. shows the framework of ORB-SLAM. For the first time, they proposed a method for position recognition using ORB features [18] based on Bag-of-Words (BoW) [19] technology. ORB feature is based on BRIEF (Binary Robust Independent Elementary Features) [20] descriptor. Features from Accelerated Segment Test (FAST) [21] key point detector, allows real-time performance without GPUs, providing good invariance to changes in viewpoint and illumination. Monocular cameras are proposed for ORB-SLAM for large-scale environments; this approach has demonstrated superior performance. Afterward, ORB-SLAM2 was extended from monocular cameras to stereo and RGB-D cameras [22].

2.2 Direct Method

Different from feature-based methods, the direct method does not rely on manually designed sparse features, but rather builds an optimization problem that estimates camera motions directly from pixel information (usually photometric errors). The direct method eliminates the time to extract features at the cost of making the scale of the optimization problem much larger than the feature-based method.

Newcombe et al. proposed a density tracking and mapping (DTAM) system [23] that computes a dense depth map for each keyframe by minimizing global, spatially regular energy functionals. The pose of the camera is determined by directly aligning the entire image with a depth map. This method is computationally intensive and only possible through GPU.

In order to reduce the amount of calculation, Forster et al. presented a Semi-Direct Visual Odometry (SVO) [24] algorithm using feature-correspondence. Feature extraction is only required when a keyframe is selected to initialize new 3D points. On an embedded computer MAVs SVO algorithm runs at more than 50 frames per second. Engel et al. then improved SVO by introducing Large-Scale Direct Monocular SLAM (LSD-SLAM) [25] which can run in large-scale environments with CPU. LSD-SLAM uses direct image alignment coupled with a filtering-based estimation of semi-dense depth maps, while the global map is represented as a pose graph consisting of keyframes as vertices with 3D similarity transforms as edges, incorporating changing scale of the environment and allowing to detect and correct accumulated drift. Engel et al. further improved the direct method and proposed Direct Sparse Odometry (DSO). DSO combines photometric errors with geometric errors and optimizes all the model parameters jointly. The demonstrated performance shows robustness in some featureless environments.

3. Visual Odometry with Deep Learning

Feature-based methods use various feature detectors to detect salient feature, for example FAST [21], SURF (Speeded Up Robust Features) [26], BRIEF [20] and Harris [27] corner detectors. The feature point tracker is then used to track these feature points in the framework of the next sequence; the most commonly used tracker is the KLT tracker [28, 29]. The output from the tracker is optical flow, after that, the camera parameters proposed by Nister [30] can be used to estimate self-motion. This general approach of detecting feature points and tracking them is followed by most works as is the case in [31, 32]. More recent works in this area employed the PTAM [2] approach, as in [33–35].

The direct method of visual odometry relies directly on the pixel intensity values in the image, and minimizes errors in sensor space, avoiding feature matching and tracking subsequently. However, these methods require planarity assumptions. Early direct monocular SLAM methods, such as Jin et al. [36], Molton et al. [37] and Silveira et al. [38] used filtering algorithms from Structure from Motion (SfM), while Pretto et al. [39] and Krizhevsky [40] used nonlinear least-squares estimation. Other methods, such as DTAM [23] require a lot of GPU parallelism.

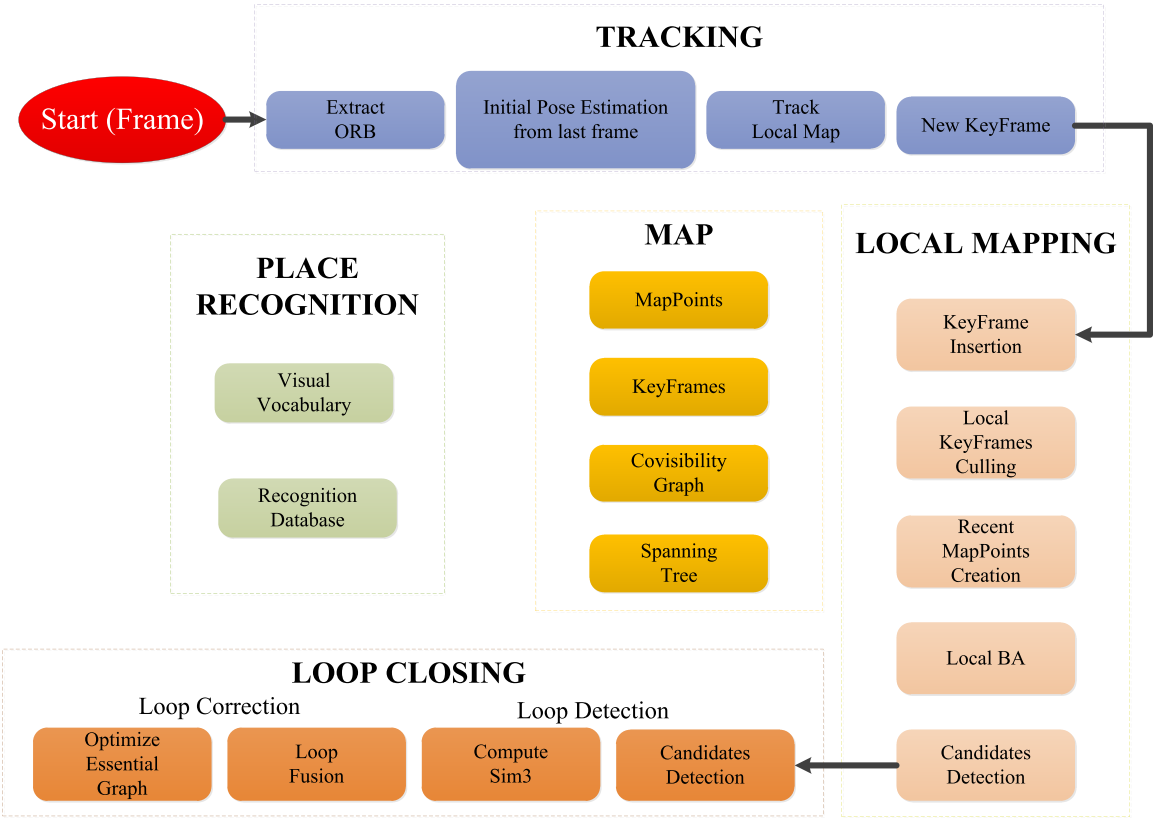

Fig. 1. ORB-SLAM [22] system overview.

In order to reduce this heavy computational demand, many researchers attempt to apply deep learning methods on the OV problem, their work can be divided into supervised methods and unsupervised methods.

3.1 Supervised Methods

Vikram et al. [10] presented a DeepVO framework for analyzing monocular VO. The framework consists of two parallel AlexNet [40] and concatenating at the end of the final convolutional layer to generate fully connected layers. They tested DeepVO using in known environments, with data segregated into training and testing sequence. Results demonstrated a competitive performance. Costante et al. [41] presented a CNNs, which parallel learning of dense optical flow [42] extracted from the global and local image.

Wang et al. [14] presented an end-to-end framework for monocular VO by using deep Recurrent Convolutional Neural Networks (RCNNs), which is composed of CNN based feature extraction and RNN based sequential modeling. Firstly, an image is fed into CNN to produce an effective feature for the monocular VO, then the learned feature is passed through an RNN for sequential learning. Experiments on the KITTI [43] VO dataset show competitive performance to state-of-the-art methods. The authors then extended the work by including uncertainty estimation, and evaluation of mobile robots, flying robots and human motion [44]. Melekhov et al. [45] also presented a relative camera pose estimation system with CNN.

Similar to the framework of Wang, Turan et al. [46] proposed a monocular visual odometry (VO) method for endoscopic capsule robot operations. The proposed Deep learning network consists of three inception layers and two LSTM layers concatenated sequentially.

3.2 Unsupervised Methods

Zhou et al. [11] presented an unsupervised learning framework with view synthesis as the supervisory signal. The method uses single-view depth and multi-view pose networks, with a loss based on warping nearby views to the target using the computed depth and pose. This system cannot recover the scale from learning monocular images.

Inspired by Zhou et al. [11], Vijayanarasimhan et al. [47] proposed SfM-Net, a geometry-aware neural network for motion estimation in videos that decomposes frame-to-frame pixel motions in

terms of scene and object depth, camera motion and 3D object rotations and translations. The model can be trained with various degrees of supervision: self-supervised by the reprojection photometric error (completely unsupervised), supervised by ego-motion (camera motion), or supervised by depth (e.g., as provided by RGBD sensors).

Mahjourian et al. [48] also focus on learning depth and ego-motion from monocular video, by combining 3D-based loss with 2D losses, based on photometric quality of frame reconstructions using estimated depth and ego-motion from adjacent frames. They also incorporate validity masks to avoid penalizing areas in which no useful information exists. Nguyen et al. proposed a hybrid approach, which combines deep learning and feature-based methods; they use features to compute the homography estimates. The network architecture is based on VGGNet [49]. Zhan et al. [50] proposed a parallel CNNs frame, experiments show jointly training for single view depth and VO improves depth prediction because of the additional constraint imposed on depths and achieves competitive results for VO.

Li et al. [51] proposed a monocular visual odometry (VO) system called UnDeepVO to estimate the 6-DoF pose of a monocular camera and the depth of its view by using deep neural networks, which use stereo image pairs to recover the scale. They were tested by using consecutive monocular images. The experiments on KITTI [43] dataset show that UnDeepVO achieves good performance in terms of pose accuracy.

4. Loop Closure with Deep Learning

The goal of loop closure detection is to give the robot the ability to recognize the same scene. In other words, the robot can tell whether it has been to the place or not [53–55]. This issue has always been one of the biggest obstacles to large-scale SLAM and recover from critical errors. Another problem arising is perceptual aliasing [56, 57]. Two different places are considered to be the same. This represents a problem even when the camera is used as a sensor due to the repetitive nature of the environment, e.g. corridor, similar architectural elements or areas with lots of bushes. A good closed-loop detection method cannot return any false positives and must obtain the least false negatives.

Ho et al. [52] used a similarity matrix to code the relationships of resemblance between all the possible pairs in captured images. They demonstrated by means of a single value decomposition that it is possible to detect loop closures, despite of the presence of repetitive and visually ambiguous images. Eade et al. [56] presented a unified method to recover from tracking failures and detect loop closures in the problem of monocular visual SLAM in real time. They also proposed a system called GraphSLAM [57] where each node stores landmarks and maintains estimations of the transformations relating nodes using ORB-SLAM [17] employed BoW [19] to detect loop.

Due to the great development and success of convolutional neural networks and deep learning in the area of computer vision [58, 59] a recent trend in autonomous robots is to exploit learned features instead of hand-crafted traditional features to tackle visual problems, especially loop closure detection problem for visual SLAM systems.

4.1 Supervised Method

Several research groups have tried to use pre-trained CNN models as feature generators to obtain whole image representations and demonstrated this on various of datasets. They concluded that CNN features are more robust regarding viewpoint, illumination and scale variations of the environment [49, 60–62].

Naseer et al. built a Caffe model by GoogLenet [63] and Alexnet [41] which they employed for detecting loop closure across seasons. The model is pre-trained on Places [64] database. In addition, a directed data association graph over the similarity matrix to leverage sequential information has been build. In [65], Bai et al. also used CNNs to extract features, and tried to improve the real-time performance of the loop closure detection using a feature compression method. Further they used O-SeqCNNSLAM [66] as image descriptors, which is based on SeqCNNSLAM, enables real-time performance and online parameters adjustment.

Instead of using features directly, Zhang et al. [67] pre-processed the CNN features by a principal component analysis (PCA) and whitening step; the framework is shown in Fig. 2. By doing so, high-dimension features are projected into a lower dimensional space, which makes the detecting process more efficient.

4.2 Unsupervised Method

Gao et al. [68] presented an unsupervised method, in which a stacked auto-encoder is employed to

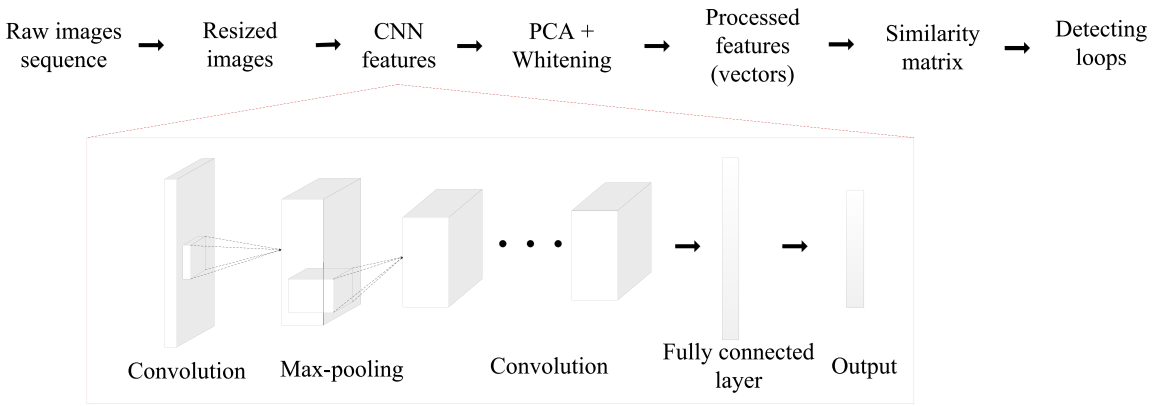

Fig. 2. The architecture of Zhang's [67] Method.

learn features, modify the object function of traditional auto-encoders by adding the denoising, sparsity and continuity cost item into it, and then evaluate the effect of corruption by experiments. Also based on autoencoder, the method in [69] inflict random noise on input data, and utilizes the geometric information and illumination invariance provided by histogram of oriented gradients (HOG) [70] forcing the encoder to reconstruct a HOG descriptor.

Based on the stacked denoising auto-encoder (SDA), Gao et al. [13] proposed a multi-layer neural network that autonomously learns a compressed representation from the raw input data in an unsupervised way.

Based on SDA, Wang et al. [71] proposed a graph-regularization stacked denoising auto-encoder (G-SDA) network and the manifold learning graph regularization structure. Compared with the bag-of-words (BoW) method, the OpenFABMAP algorithm, and the traditional SDA method, their method achieves superior performances.

5. Develop trends

The geometry-based SLAM method has achieved high precision and real-time, but these algorithms are tending to fail under different lighting conditions, due to the movement of people or objects, the emergence of feature-free regions, day and night transitions or any other unforeseen circumstances.

With the power of deep learning shown in a variety of visual tasks, people's attention has gradually turned to deep learning solutions. In addition, a visual SLAM system with learning or adaptive capabilities is a factor which is worth further exploration. The success of deep learning is centered around long-term training on supercomputers and the use of dedicated GPU hardware to achieve one-off results. One of the challenges faced by SLAM researchers is how to provide enough computing power in embedded systems. A bigger and more important challenge is online learning and adaptation, which will be essential for any future long-term visual SLAM system.

Most existing networks tend to train a large number of labeled data, but it is not always possible to guarantee the existence of a suitable data set. A semi-supervised network that uses a small set of labeled data and a large amount of unlabeled data to train is an important direction for the future development of visual SLAM.

The visual SLAM system usually runs in an open world, where new objects and scenes can be encountered. But so far, deep networks are often trained for closed world scenarios. A deep network tends to perform well in its trained datasets, and tend to fail in untrained datasets. A major challenge is to enable deep learning networks for lifelong learning visual SLAM systems.

6. Conclusion

The knowledge of geometry-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 datasets is the 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.

Conflict of interest statement. None declared.

References

1. Robotic Mapping SC. *Exploration*. Berlin: Springer, 2009.
2. Klein G, Murray D. Parallel tracking and mapping for smallAR workspaces. In: *Proceedings of the 2007 6th IEEE and ACM*

International Symposium on Mixed and Augmented Reality. Nara, Japan, 2007, 1–10
3. Geiger A, Lenz P, Urtasun R. Are we ready for autonomous driving? The KITTI vision benchmark suite. In: *2012 IEEE Conference on Computer Vision and Pattern Recognition. Providence, RI, USA*, 2012, 3354–61
4. Artieda J, Sebastian JM, Campoy P, *et al*. Visual 3-D SLAM from UAVs. *J Intell Robot Syst* 2009; **55**:299.
5. Hong S, Chung D, Kim J, *et al*. In-water visual ship hull inspection using a hover-capable underwater vehicle with stereo vision. *J Field Robot* 2019; **36**:531–46.
6. Grasa OG, Bernal E, Casado S, *et al*. Visual SLAM for handheld monocular endoscope. *IEEE T Med Imaging* 2014; **33**: 135–46.
7. Schöps T, Engel J, Cremers D. Semi-dense visual odometry for AR on a smartphone. *IEEE International Symposiumon Mixed and Augmented Reality (ISMAR), Munich, Germany* 2014; **2014**:145–50.
8. Parkhi OM, Vedaldi A, Zisserman A. Deep face recognition. *Proceedings of the British Machine Vision Conference (BMVC),Swansea, UK* 2015; 1–12.
9. Amodei D, Ananthanarayanan S, Anubhai R, *et al*. Deep Speech 2: end-to-end speech recognition in English and Mandarin. In: *Proceedings of the 33rd International Conference on Machine Learning, New York, USA*, 2016, 173–82
10. Mohanty V, Agrawal S, Datta S, *et al*. DeepVO: A deep learning approach for monocular visual odometry. arXiv: 1611.06069 [cs.CV], 2016.
11. Zhou T, Brown M, Snavely N, *et al*. Unsupervised learning of depth and ego-motion from video. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Honolulu,HI, USA* 2017; **2017**:6612–9.
12. Naseer T, Ruhnke M, Stachniss C, *et al*. Robust visual SLAM across seasons. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany* 2015; **2015**:2529–35.
13. Gao X, Zhang T. Unsupervised learning to detect loops using deep neural networks for visual SLAM system. *Auton Robot* 2017; **41**:1–18.
14. Wang S, Clark R, Wen H, *et al*. DeepVO: towards end-to-end visual odometry with Dee recurrent convolutional neural networks. *IEEE International Conference on Robotics and Automation (ICRA), Singapore* 2017; **2017**:2043–50.
15. Davison AJ, Reid ID, Molton ND, *et al*. MonoSLAM: realtime single camera SLAM. *IEEE T Pattern Anal* 2007; **29**:1052–67.
16. Triggs B, McLauchlan PF, Hartley RI, *et al*. Bundle adjustment—a modern synthesis. In: *Vision Algorithms: Theory and Practice. Corfu, Greece*, 1999, 298–372
17. Mur-Artal R, Tardós JD. ORB-SLAM2: an open-source SLAM system for monocular, stereo, and RGB-D cameras. *IEEE TRobot* 2017; **33**:1255–62.
18. Engel J, Schöps T, Cremers D. LSD-SLAM: large-scale direct monocular SLAM. In: *Computer Vision – ECCV 2014. Zurich, Switzerland*, 2014, 834–49
19. Mur-Artal R, Montiel JMM, Tardós JD. ORB-SLAM: a versatile and accurate monocular SLAM system. *IEEE T Robot* 2015; **31**:1147–63.
20. Rublee E, Rabaud V, Konolige K, *et al*. ORB: an efficient alternative to SIFT or SURF. In: *2011 International Conference on Computer Vision. Barcelona, Spain*, 2011, 2564–71.
21. Galvez-López D, Tardos JD. Bags of binary words for fast place recognition in image sequences. *IEEE T Robot* 2012; **28**:1188–97.
22. Calonder M, Lepetit V, Strecha C, *et al*. BRIEF: Binary robust independent elementary features. In: Computer Vision – ECCV 2010. Heraklion, Greece, 2010, 778–92.Rosten E, Drummond T. Machine learning for high-speed corner detection. In: *Computer Vision – ECCV*, Vol. 2006. Austria: Graz, 2006, 430–43
23. Newcombe RA, Lovegrove SJ, Davison AJ. DTAM: dense tracking and mapping in real-time. In: *2011 International Conference on Computer Vision, Barcelona, Spain*, 2011, 2320–7
24. Forster C, Pizzoli M, Scaramuzza D. SVO: fast semi-direct monocular visual odometry. *IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China* 2014; **2014**:15–22.
25. Engel J, Sturm J, Cremers D. Semi-dense visual odometry for a monocular camera. In: *2013 IEEE International Conference on Computer Vision. Sydney, Australia*, 2013, 1449–56
26. Bay H, Tuytelaars T, Van Gool L. SURF: speeded up robust features. In: *Computer Vision – ECCV 2006. Graz, Austria*, 2006, 404–17
27. Harris C, Stephens M. A combined corner and edge detector. *Alvey Vision Conference, Manchester, UK* 1988; 147–52.
28. Tomasi C, Kanade T. *Tracking of point features*. In: Technical report. Carnegie Mellon University, 1991.
29. Shi J, Tomasi C. Good features to track. In: *1994 Proceedings of 1994 IEEE Conference on Computer Vision and Pattern Recognition, Seattle,WA, USA*, 1994, 593–600
30. Nister D. An efficient solution to the five-point relative pose problem. *IEEE T Pattern Anal* 2004; **26**:756–70.
31. Matthies L. Dynamic stereo vision. Ph.D. Thesis. In: *Carnegie Mellon University*, 1989
32. Johnson AE, Goldberg SB, Cheng Y, *et al*. Robust and efficient stereo feature tracking for visual odometry. In: *2008 IEEE International Conference on Robotics and Automation, Pasadena, CA, USA*, 2008, 39–46
33. Blösch M, Weiss S, Scaramuzza D, *et al*. *Vision based MAV navigation in unknown and unstructured environments. In: 2010 IEEE International Conference on Robotics and Automation*. Anchorage, AK: USA, 2010, 21–8.
34. Weiss S, Achtelik MW, Lynen S, *et al*. Monocular vision for long-term micro aerial vehicle state estimation: a compendium. *J Field Robot* 2013; **30**:803–31.
35. Kneip L, Chli M, Siegwart R. Robust real-time visual odometry with a single camera and an IMU. In: *Proceedings of the British Machine Vision Conference 2011,Dundee, UK*, 2011, 1–16
36. Jin H, Favaro P, Soatto S. A semi-direct approach to structure from motion. *Visual Comput* 2003; **19**:377–94.
37. Molton N, Davison AJ, Reid ID. Locally planar patch features for real-time structure from motion. In: *Proceedings of the British Machine Vision Conference 2004, London, UK*, 2004, 1–10
38. Silveira G, Malis E, Rives P. An efficient direct approach to visual SLAM. *IEEE T Robot* 2008; **24**:969–79.
39. Pretto A, Menegatti E, Pagello E. Omnidirectional dense large-scale mapping and navigation based on meaningful triangulation. In: *2011 IEEE International Conference on Robotics and Automation, Shanghai, China*, 2011, 3289–96
40. Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep convolutional neural networks. In: *Proceedings of the 25th International Conference on Neural Information Processing Systems, Lake Tahoe, NV*, 2012, 1097–105
41. Costante G, Mancini M, Valigi P, *et al*. Exploring representation learning with CNNs for frame-to-frame ego-motion estimation. *IEEE Robot Autom Lett* 2016; **1**:18–25.

42. Brox T, Bruhn A, Papenberg N, *et al*. High accuracy optical flow estimation based on a theory for warping. In: *Computer Vision - ECCV 2004, Prague, Czech Republic*, 2004, 25–36
43. Geiger A, Lenz P, Stiller C, *et al*. Vision meets robotics: the KITTI dataset. *Int J Robot Res* 2013; **32**:1231–7.
44. Wang S, Clark R, Wen H, *et al*. End-to-end, sequence-tosequence probabilistic visual odometry through deep neural networks. *Int J Robot Res* 2018; **37**:513–42.
45. Melekhov I, Ylioinas J, Kannala J, *et al*. Relative camera pose estimation using convolutional neural networks. In: *Advanced Concepts for Intelligent Vision Systems. Antwerp, Belgium*, 2017, 675–87
46. Turan M, Almalioglu Y, Araujo H, *et al*. Deep EndoVO: a recurrent convolutional neural network (RCNN) based visual odometry approach for endoscopic capsule robots. *Neurocomputing* 2018; **275**:1861–70.
47. Vijayanarasimhan S, Ricco S, Schmid C, *et al*. *SfM-Net: learning of structure and motion from video. arXiv*:1704.**07804** [cs.CV], 2017.
48. Mahjourian R, Wicke M, Angelova A. Unsupervised learning of depth and ego-motion from monocular video using 3D geometric constraints. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA* 2018; 5667–75.
49. Simonyan K, Zisserman A. *Very deep convolutional networks for large-scale image recognition. arXiv*:1409.**1556** [cs.CV], 2014.
50. Zhan H, Garg R,Weerasekera CS, *et al*. Unsupervised learning of monocular depth estimation and visual odometry with deep feature reconstruction. arXiv:1803.03893 [cs.CV], 2018.
51. Li R, Wang S, Long Z, *et al*. UnDeepVO: monocular visual odometry through unsupervised deep learning. *IEEE International Conference on Robotics and Automation (ICRA), Brisbane, Australia* 2018; **2018**:7286–91.
52. Ho KL, Newman P. Detecting loop closure with scene sequences. *Int J Comput Vision 2007; 74:261–86*. Clemente LA, Davison AJ, Reid ID, *et al*. Mapping large loops with a single hand-held camera. *In: Robotics: Science and Systems*. Atlanta, GA, USA, 2007.
53. Mei C, Sibley G, Cummins M, *et al*. RSLAM: a system for largescale mapping in constant-time using stereo. *Int J Comput Vision* 2011; **94**:198–214.
54. Angeli A, Doncieux S, Meyer J-A, *et al*. Real-time visual loop-closure detection. In: *2008 IEEE International Conference on Robotics and Automation, Pasadena, CA, USA*, 2008, 1842–7
55. CumminsM,Newman P. FAB-MAP: probabilistic localization and mapping in the space of appearance. *Int J Robot Res* 2008; **27**:647–65.
56. Eade E, Drummond T. Unified loop closing and recovery for real time monocular SLAM. In: *Proceedings of the British Machine Vision Conference 2008, Leeds, UK*, 2008, 1–10
57. Eade E, Fong P, Munich ME. Monocular graph SLAM with complexity reduction. In: *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, Taipei, Taiwan*, 2010, 3017–24
58. Bengio Y, Courville A, Vincent P. Representation learning: a review and new perspectives. *IEEE T Pattern Anal* 2013; **35**:1798–828.
59. LeCun Y, Bengio Y, Hinton G. Deep learning. *Nature* 2015; **521**:436–44.
60. Sünderhauf N, Shirazi S, Jacobson A, *et al*. Place recognition with ConvNet landmarks: viewpoint-robust, condition-robust, training-free. *Robotics: Science and Systems. Rome, Italy* 2015.
61. Hou Y, Zhang H, Zhou S. Convolutional neural networkbased image representation for visual loop closure detection. In: *2015 IEEE International Conference on Information and Automation, Lijiang, China*, 2015, 2238–45
62. Sünderhauf N, Shirazi S, Dayoub F, *et al*. On the performance of ConvNet features for place recognition. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany* 2015; **2015**:4297–304.
63. Szegedy C, Liu W, Jia Y, *et al*. Going deeperwith convolutions. In: 2015 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA*, 2015, 1–9.
64. Zhou B, Lapedriza A, Xiao J, *et al*. Learning deep features for scene recognition using places database. In: *Advances in Neural Information Processing Systems (NIPS). Montreal, Canada*, 2014, 487–95
65. Bai D, Wang C, Zhang B, *et al*. Matching-range-constrained real-time loop closure detection with CNNs features. *Robot Biomim* 2016; **3**:15.
66. Milford MJ, Wyeth GF. SeqSLAM: visual route-based navigation for sunny summer days and stormy winter nights. In: *2012 IEEE International Conference on Robotics and Automation, Saint Paul, MN, USA*, 2012, 1643–9
67. Zhang X, Su Y, Zhu X. Loop closure detection for visual SLAM systems using convolutional neural network. *In: 2017 23rd International Conference on Automation and Computing (ICAC)*. Huddersfield, UK, 2017, 1–6
68. Gao X, Zhang T. Loop closure detection for visual SLAM systems using deep neural networks. *34th Chinese Control Conference (CCC). Hangzhou, China* 2015; **2015**:5851–6.
69. Merrill N, Huang G. *Lightweight unsupervised deep loop closure. arXiv*:1805.**07703** [cs.RO], 2018.
70. Dalal N, Triggs B. Histograms of oriented gradients for human detection. In: *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. San Diego, CA: USA, 2005
71. Wang Z, Peng Z, Guan Y, *et al*. Manifold regularization graph structure auto-encoder to detect loop closure for visual SLAM. *IEEE Access* 2019; **7**:59524–38.