Monocular Dense SLAM with Consistent Deep Depth Prediction

Feihu Yan[1], Jiawei Wen[1], Zhaoxin Li[2], and Zhong Zhou[1(✉)]

[1] State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing 100191, China
zz@buaa.edu.cn
[2] Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China

Abstract. Monocular simultaneous localization and mapping (SLAM) that using a single moving camera for motion tracking and 3D scene structure reconstruction, is an essential task for many applications, such as vision-based robotic navigation and augmented reality (AR). However, most existing methods can only recover sparse or semi-dense point clouds, which are not adequate for many high-level tasks like obstacle avoidance. Meanwhile, the state-of-the-art methods use multi-view stereo to recover the depth, which is sensitive to the low-textured and non-Lambertian surface. In this work, we propose a novel dense mapping method for monocular SLAM by integrating deep depth prediction. More specifically, a classic feature-based SLAM framework is first used to track camera poses in real-time. Then an unsupervised deep neural network for monocular depth prediction is introduced to estimate dense depth maps for selected keyframes. By incorporating a joint optimization method, predicted depth maps are refined and used to generate local dense submaps. Finally, contiguous submaps are fused with the ego-motion constraint to construct the globally consistent dense map. Extensive experiments on the KITTI dataset demonstrate that the proposed method can remarkably improve the completeness of dense reconstruction in near real-time.

Keywords: Dense Mapping · Visual SLAM · Monocular Depth Prediction.

1 Introduction

Taking advantage of the universality and simplicity of camera sensors, monocular SLAM [2, 24], which typically performs localization while building a 3D map of the surrounding environment simultaneously by using only a single camera, has

Supported by the National Key Research and Development Program of China under Grant 2018YFB2100601, and National Natural Science Foundation of China under Grant (61872023, 61702482).

been extensively studied in the past two decades and has been deployed in various applications, including online 3D modeling [16], AR [11, 12], and autonomous navigation [4].

SLAM methods can be basically classified into feature-based [14] and direct approaches [6, 5]. Feature-based methods normally extract a set of point features from images, which are used to steadily track camera poses and reconstruct sparse 3D point clouds. Direct method utilize images directly without any abstraction and can generate semi-dense maps [5, 6], but the sensitivity to luminosity changes makes them not as robust as feature-based methods in many application scenarios. However, neither sparse nor semi-dense 3D maps are adequate for tasks like obstacle avoidance or interaction of virtual and physical objects.

Unlike RGB-D SLAM systems [15, 9] that can directly obtain dense depth information from depth sensors for dense reconstruction, it is challenging for monocular SLAM methods to estimate a consistent dense map. One of the main reasons is that the commonly used multi-view stereo method is sensitive to the low-textured and non-Lambertian surface, which leads to incomplete depth estimation. Based on the plane assumption, some works utilize superpixels [3, 19] or depth interpolation [20] to generate dense maps, which improve the 3D reconstruction completeness for planar environments. Nonetheless, these works are limited to non-planar scenes and tend to over-smooth the surface details.

More recently, with the rapid development of deep learning techniques, deep neural networks [8] did dramatically boost the performance of depth prediction from monocular images (depth-from-mono). Subsequently, some works try to combine traditional SLAM systems with deep depth prediction networks [18, 10, 21]. However, most of these works focus on small indoor scenes and are not generalized to large-scale outdoor environments. Moreover, most methods use the depth prediction as the prior fed into the SLAM system, or directly use it in the RGB-D SLAM framework, without considering the introduced additional uncertainty introduced, which makes the reconstruction heavily dependent on the accuracy of the depth estimation.

In this paper, we propose a novel dense mapping method for monocular SLAM with consistent deep depth prediction. Our method utilizes the classic feature-based SLAM framework, ORB-SLAM2 [14], to track camera poses in real-time. When one new keyframe is created, it is fed into an unsupervised deep neural network [8] to predict the corresponding depth map, which will be refined and used in the subsequent process to generate the local 3D submap. Finally, contiguous submaps will be fused with the ego-motion constraint to construct a globally consistent dense map. The main contributions of this work are summarized as follows: 1) We present a novel dense mapping method for monocular visual SLAM, which integrates the deep depth prediction with the feature-based SLAM framework. 2) We propose a joint optimization method from 2D and 3D aspects to deal with the uncertainty introduced by the predicted depth, and generate a globally consistent dense map with the ego-motion constraint.

2 Related Work

2.1 Monocular Visual SLAM

In the past two decades, monocular SLAM has been extensively investigated and a large variety of advanced algorithms have been proposed. There are two main categories, feature-based and direct methods. The feature-based methods need to extract feature points first, which ensures robustness but also leads to extremely sparse reconstruction. Typically, ORB-SLAM2 [14] is one of the most widely used feature-based SLAM frameworks, which contains full capabilities including loop closing for a complete SLAM system. Directly using the raw pixels without any abstraction, direct methods have the ability to provide more expressive semi-dense maps, however, they have to spend extra efforts to deal with photometric changes [5].

In general, compared to direct methods, feature-based SLAM systems are not sensitive to photometric changes and perform better when the camera is moving forward, which makes them more suitable for many outdoor scenarios. Considering robustness, practicability, and scalability, we choose the widely used ORB-SLAM2 [14] as the basic framework to track poses of the moving camera, meanwhile, its loop closing thread can correct the accumulated drift and provide help for the construction of a globally consistent map.

2.2 Dense Mapping

Most dense SLAM systems typically build dense maps with available depth information using RGB-D cameras, such as [15, 9]. However, since the depth camera can only provide reliable measurements in a limited range, the applicable working scenarios of these methods are limited, usually indoor scenes.

Some researchers have investigated how to obtain dense maps using a single monocular camera and proposed many charming works. Newcombe et al. [16] presented a dense SLAM system that generates smooth depth estimates by a non-convex optimization process. This system needs GPU to optimize the variational model, and the high computational cost limits its availability in large-scale environments. Concha et al. [3] proposed a dense mapping approach, which combines semi-dense maps [6] with superpixels. This work performs well in indoor scenes where low-texture regions are usually flat. Teixeira et al. [19] proposed a dense reconstruction method for small unmanned aerial vehicles (UAVs), which combines ORB-SLAM [13] with superpixels to provide a local semi-dense reconstruction in real-time.

Inspired by [3], Xue et al. [22] proposed a real-time monocular dense mapping method, which replaces the superpixel method with another efficient homogeneous region detector. Wang et al. [20] proposed a monocular dense mapping method for UAV navigation, which uses the quadtree-based pixel selection to accelerate the mapping.

Although these works have achieved amazing experiments, building denser maps is still a challenging task for large-scale outdoor scenes, partly due to the widespread existence of low-textured and non-planar areas.

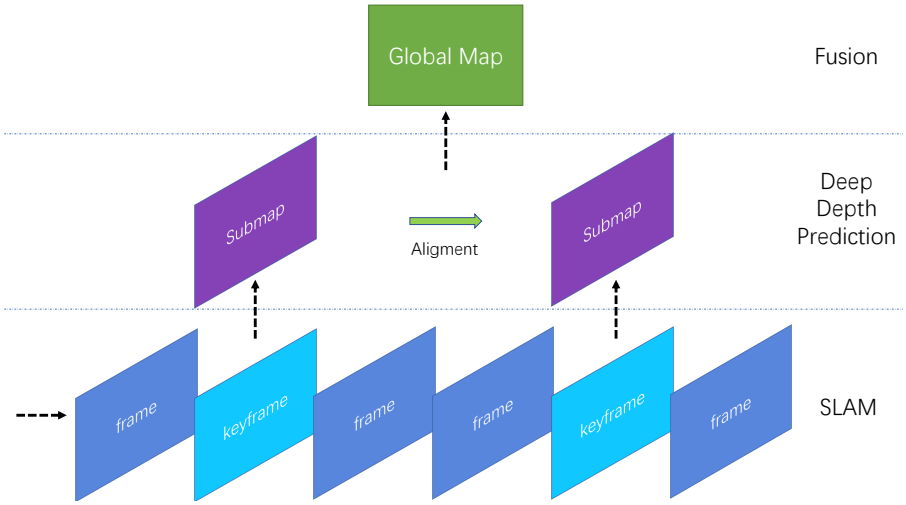

Fig. 1. System Overview. Camera poses are estimated by the SLAM system. Then submaps are generated from selected keyframs with refined deep depth prediction. Finally, optimized submaps are fused into a globally consistent dense map.

2.3 SLAM with Deep Depth Prediction

In recent years, with the rapid development of deep learning technology, convolutional neural networks (CNNs) have been widely used in monocular depth estimation. Some researchers have tried to fuse deep depth prediction with traditional SLAM systems. As the efficiency and accuracy of depth prediction have been significantly improved, this fusion has become a trend.

Tateno et al. [18] proposed a breakthrough work, which combines CNN-based depth prediction with LSD-SLAM [6] to obtain dense maps. They also proposed an extension that fuses semantic labels with the dense map. Ji et al. [10] presented a depth fusion framework, which exploits the sparse depth estimation from ORB-SLAM [13] and the CNN-inferred depth to generate a dense reconstruction. These two methods use the direct method and the feature point method as the SLAM framework, and the application scenarios are mostly indoor scenes. Wang et al. [21] proposed a surfel-based dense mapping method, which can fuse dense maps for large-scale outdoor scenes. When using a monocular camera, ORB-SLAM2 [14] in RGB-D mode is used to track camera poses with the deep depth prediction. In order to gain the run-time efficiency, surfels are used to represent the map, but also reduce the density of the point cloud.

3 System Overview

The pipeline of our work is illustrated in Fig. 1. We first use the state-of-the-art visual SLAM system, ORB-SLAM2 [14], to estimate the camera poses and

extract keyframes. Then an unsupervised deep neural network[8] is introduced to predict a dense depth map for each keyframe. Refined depth maps are used to generate submaps from keyframes. Finally, contiguous submaps are fused to obtain a globally consistent dense map.

More specifically, when one new frame F_i comes, it is firstly tracked with respect to the reference keyframe K_r. If it is too far from the reference keyframe or the visual change conditions are met [14], F_i is chosen to generate a new keyframe. Every new keyframe is simultaneously fed into the deep neural network to estimate a dense depth map D_i.

Given the camera pose R, t and the dense depth estimation D_i of each keyframe, we aim to automatically generate a consistent dense map in near real-time. To achieve this, we first refine the depth map D_i using a joint optimization method. The local 3D submap S_i for each keyframe K_i can be obtained using the refined depth map $\hat{D}_i$. Then a classical point cloud registration method is used to estimate the spatial relationship between contiguous local submaps. Finally, 3D point clouds are fused with the camera ego-motion constraint to obtain a consistent dense map. An example of dense mapping is shown in Fig. 2.

4 Local Mapping with Depth Refinement

Given the depth maps predicted by the deep neural network [8], we can easily construct local 3D submaps with intrinsic camera parameters. However, the depth value predicted by the network contains more noise than the depth measurement obtained by the depth sensor. To refine the depth prediction, we mainly consider dealing with the 2D image areas that are likely to cause uncertainty in the depth prediction; in addition, we also need to filter out outliers in the 3D submaps.

4.1 2D Image Analysis

Intuitively, there are three types of image areas that mainly cause uncertainty in the depth prediction, including image boundaries, object contours, and pixels far away from the optical center of the camera.

Image Boundary In the process of depth prediction, the deep neural network [8] learns to predict the pixel-level depth by incorporating an inbuilt left-right consistency check. Given the baseline d, the camera focal length f, and the predicted image disparity d for per pixel, the depth z can be obtained as follows,

$$z = bf/d \tag{1}$$

Intuitively, due to the lack of left-right consistency in the areas located at the image boundary, the depth estimation is more likely to produce high-uncertainty predictions.

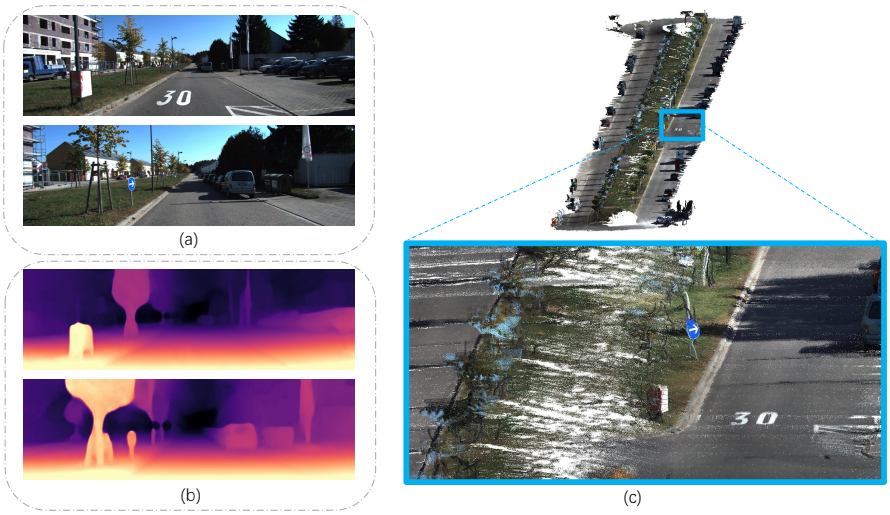

Fig. 2. The proposed method achieves dense mapping for monocular SLAM by fusing the deep depth prediction to recover a consistent 3D dense map. (a) shows two example keyframes extracted from KITTI sequence 06. (b) illustrates the corresponding depth map predicted by the unsupervised deep neural network [8]. (c) shows the global dense map (top) and details of the region labeled at blue wireframes (bottom).

To ensure efficient depth estimation, we first crop the depth image according to the visual overlap area. In other words, the depth estimation near the image boundary will be discarded, which also reduces the computing consumption. As illustrated in Fig. 2 (c), the final dense map discards the boundary parts while keeping the middle regions.

Object Contour We observe that, as shown in Fig. 2 (b), the depth near the contour of the object is typically over-smoothed, such as the contours of trees and cars in the figure. Therefore, we regard the image area near the contour of the object as another factor that easily leads to uncertainty for depth estimation.

We use a filter-based method to optimize the depth information. More specifically, we use a Gaussian weight function, which calculates the corresponding weight to refine the predicted depth. In order to ensure the distinguishability of the boundaries of different objects, we combine the information of color and depth difference within a small neighborhood $S_{p_i} = \{p_j\}$ around a pixel p_i into a multilateral filtering process. The refined depth can be obtained as follows,

$$\hat{D_{p_i}} = \frac{1}{W_{p_i}} \sum_{p_j \in S_{p_i}} G_{\sigma_s}(\|p_i - p_j\|) G_{\sigma_c}\left(\left\|I_{p_i} - I_{p_j}\right\|\right) G_{\sigma_d}\left(\left\|D_{p_i} - D_{p_j}\right\|\right) D_{p_i} \quad (2)$$

where i and j are pixel indexes, I_p is the color, and D_p is the corresponding depth in the predicted depth map D. G denotes the Gaussian filter kernel, while

the parameters σ_e, σ_c, and σ_d are used to adjust the spatial similarity, the color similarity, and the depth similarity, respectively. W_{p_i} is used for normalization.

According to Eq. 2, areas with sharp color or depth changes in the depth map will be enhanced, while the other areas will be smoothed.

Far Points In stereo vision, if a 3D point is farther from the camera, the uncertainty of its depth estimation will typically be greater. Similarly, since the depth estimation network relies on the parallax information from the left-right image pairs, which could cause high uncertainty for far points. Thus, we need to detect the pixels corresponding to the far point in the image.

As suggested in ORB-SLAM2 [14], keypoints will be classified as close or far when using stereo cameras. More specifically, a stereo keypoint will be classified as the close point when its depth is less than 40 times the stereo baseline, otherwise, it is classified as a far point. In this work, we follow the strategy to detect far points and discard them when building local submaps.

4.2 3D Outlier Detection

Due to the occlusion, etc., it is difficult to avoid outliers in the predicted depth [8], which may introduce additional errors for the global mapping process. Thus, an outlier detection process is required to refine the generated submaps.

In this work, LOF (Local Outlier Factor) [1] is used to detect outliers in 3D submaps. More specifically, we calculate the LOF score for each 3D point P_i as follows,

$$LOF(P_i) = \frac{\sum_{P_j \in N_k(P_i)} \frac{lrd_k(P_j)}{lrd_k(P_i)}}{|N_k(P_i)|} \tag{3}$$

where $N_k(\cdot)$ is the k-nearest neighborhood of one 3D point and $|N_k(\cdot)|$ its size, and $lrd_k(\cdot)$ is the reciprocal of the average distance from one point to all its neighbors.

When $LOF(P_i) > 1$, it means that the local point set around P_i is sparser than its neighbor points, and P_i can be regarded as a candidate outlier.

5 Global Dense Mapping with Egomotion Constraints

It is worth noting that the unsupervised deep neural network [8] uses pairs of rectified stereo images that have the known camera baseline for training. Thus, the predicted depth contains implied scale constraint from the camera baseline, which encourages us to use the Iterative Closest Point (ICP) algorithm to align adjacent submaps and generate the global dense map.

To guarantee a consistent global map, we refine contiguous submaps according to the ego-motion estimation of corresponding keyframes from the SLAM system. On the one hand, the ego-motion obtained by SLAM is continuously optimized, and the deep prediction network will only produce the result once.

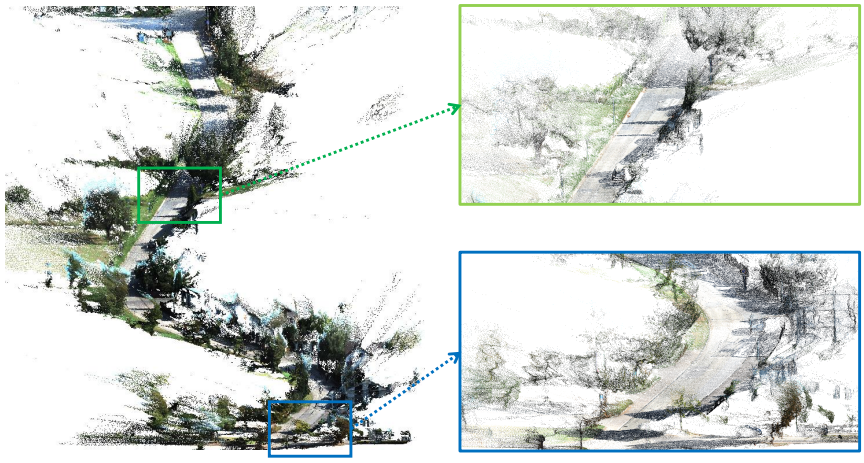

Fig. 3. Qualitative result of the dense mapping on the KITTI sequence 13. The details of the regions labeled at green and blue wireframes are also shown in the zoom-in patches (right).

On the other hand, the loop closing thread of the SLAM system can help to address the scale drift.

Inspired by [23], we propose a simple scale factor $f = t_{slam}/t_{icp}$ to guarantee a consistent scale, which is the ratio between the translational motion of the SLAM system t_{slam} and ICP t_{icp}. In contrast to [23] which refines the ego-motion and the depth map alternately, the continuously optimized camera poses of the SLAM system rather than the constant depth prediction from the deep neural network are trusted in our work. Another benefit is that, when local optimization or loop closure occurs, submaps corresponding to adjusted keyframes can be updated just simply by multiplying the updated matrix calculated from the SLAM system.

Thus, we use the scale factor to refine the depth estimation. More specifically, the scale factor is used to update the initial ICP transformation matrix T_s^{s+1} estimated between two adjacent submaps:

$$\hat{\mathbf{T}}_s^{s+1} = \begin{pmatrix} \mathbf{R}_s^{s+1} & f\mathbf{t}_s^{s+1} \\ \mathbf{0} & 1 \end{pmatrix} \tag{4}$$

Then $\hat{\mathbf{T}}_s^{s+1}$ can be used to refine the depth maps. Please refer to [23] for more technical details.

6 Evaluation

In this section, we conduct experiments to verify the effectiveness of our method on the KITTI dataset [7]. The proposed method is based on the monocular mode

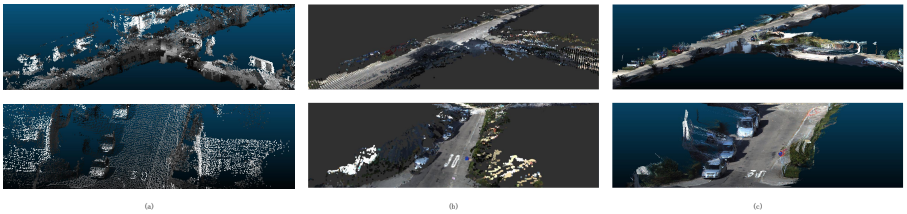

Fig. 4. Qualitative comparisons on dense mapping results of the KITTI sequence 00 using (a) Surfel-Mapping [21] (b) GEM [17] and (c) our method.

of ORB-SLAM2 [14], which is used to track camera poses, detect keyframes, and close loops. We use an unsupervised deep neural network [8] to obtain dense depth maps. Since the network performs amazingly on the KITTI dataset, it is only fine-tuned on the KITTI training set. All the experiments are carried out on a standard desktop PC with Intel Xeon CPU at 3.5GHz, 32GB of RAM, and NVIDIA GeForce GTX 1080 GPU.

6.1 Qualitative Results

Here we discuss the completeness of the final 3D map. Fig. 2 (c) shows the recovered dense map on the KITTI sequence 06 datasets using our method. The global map shown on the top is quite dense, and the zoom-in patches on the bottom perform well in fine details while keeping consistent scene structure, such as the traffic sign and the road surface. Please note that as explained in Section 4.1, the depth predictions near the image boundaries have been discarded, thus areas on both sides of the road are mostly incomplete.

Similarly, Fig. 3 demonstrates the recovered dense point cloud of the KITTI sequence 13 using our method, where the zoomed-in areas verify the density and consistency of the global map.

Qualitative comparisons are demonstrated in Fig. 4, where reconstructed dense maps are built by Surfel-Mapping [21] using stereo cameras, GEM [17] using LiDAR, and our work using a monocular camera, respectively. Our method shows significant superiority for dense mapping over previous work. Moreover, the top row shows the details of a loop closure region and our method can generate smoother local maps.

Our method also has some shortcomings. Fig. 5 shows the dense mapping result generated on the KITTI sequence 16 by our method. The details of the region labeled at green wireframes show the performance of our work when dealing with static scenes, i.e., the sign on the road could be clearly identified. However, when dealing with dynamic objects, submaps may not be aligned well in these regions. For example, in the failure case marked by red wireframes, the moving white car could not be registration successfully, which leads to 3D ghosting in the global map. In future work, we will try to introduce semantic information or object detection to deal with this problem.

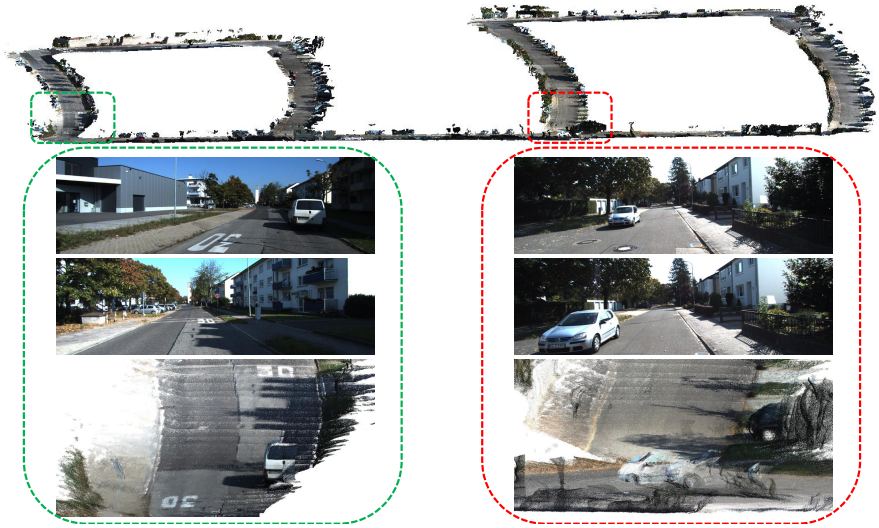

Fig. 5. Qualitative results on the KITTI sequence 16, including successful and failed cases. The global dense map generated from our work is shown on the top, while the details together with two corresponding example frames of the regions labeled at green and red wireframes are also shown in the zoom-in patches (bottom).

6.2 Quantitative Results

Table 1 shows the quantitative results on the KITTI sequence 00. Since the ground-truth point cloud is not available, we mainly report the density, completeness, and running time in this section. Note that the number of points is calculated from the final global map, while the completeness is the average ratio of pixels with valid depth estimation (the discarded ones will not be counted) for all keyframes. It demonstrates that the proposed method can generate denser maps.

Table 2 displays the average computational cost for each step. Specifically, the submap optimization takes the majority of the time, i.e. almost 3 seconds per keyframe, while the other processes could perform in real-time. Since one new keyframe will be generated when more than 20 frames have passed in the SLAM system, our work could run in near real-time (6-7 fps). Moreover, we can further reduce the number of selected keyframes to improve the efficiency of the dense mapping thread.

7 Conclusion

In this paper, we present a novel dense mapping approach for monocular SLAM that fuses both the monocular depth prediction and the camera ego-motion estimation, bridging the classic feature-based SLAM system and the deep neural network. Submaps are generated according to the refined depth prediction of

Table 1. Number of Points (K) and Average Keyframe Completeness (%).

methods	ORB-SLAM2 [14]	Surfel-Mapping [21]	Ours
points	50.6	1422.968	18563
completeness	0.05	2.2	19.85

Table 2. Average computational cost of each step for per keyframe (ms)

SLAM Tracking	Depth Prediction	Submap Optimization	Global Fusion
23	30	3000	35

keyframes, and the fusion is realized simply by aligning contiguous submaps with ego-motion constraints. Experiments on the KITTI dataset demonstrate that our method could obtain dense maps on large-scale outdoor scenes in near real-time. In the future, we plan to further refine the global point clouds and focus on dealing with dynamic objects using more semantic information.

References

1. Breunig, M.M., Kriegel, H.P., Ng, R.T., Sander, J.: Lof: Identifying density-based local outliers. SIGMOD Rec. **29**(2), 93–104 (2000)
2. Cadena, C., Carlone, L., Carrillo, H., Latif, Y., Scaramuzza, D., Neira, J., Reid, I., Leonard, J.J.: Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. IEEE Transactions on Robotics **32**(6), 1309–1332 (2016)
3. Concha, A., Civera, J.: Dense Piecewise Planar Tracking and Mapping from a Monocular Sequence. In: Proc. of The International Conference on Intelligent Robots and Systems (IROS) (2015)
4. Deng, X., Zhang, Z., Sintov, A., Huang, J., Bretl, T.: Feature-constrained active visual slam for mobile robot navigation. In: 2018 IEEE International Conference on Robotics and Automation (ICRA). pp. 7233–7238 (2018)
5. Engel, J., Koltun, V., Cremers, D.: Direct sparse odometry. IEEE Transactions on Pattern Analysis and Machine Intelligence **40**(3), 611–625 (2018)
6. Engel, J., Schöps, T., Cremers, D.: LSD-SLAM: Large-scale direct monocular SLAM. In: European Conference on Computer Vision (ECCV) (2014)
7. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In: Conference on Computer Vision and Pattern Recognition (CVPR) (2012)
8. Godard, C., Aodha, O.M., Brostow, G.J.: Unsupervised monocular depth estimation with left-right consistency. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6602–6611 (2017)
9. Hermans, A., Floros, G., Leibe, B.: Dense 3d semantic mapping of indoor scenes from rgb-d images. In: 2014 IEEE International Conference on Robotics and Automation (ICRA). pp. 2631–2638 (2014)
10. Ji, X., Ye, X., Xu, H., Li, H.: Dense reconstruction from monocular slam with fusion of sparse map-points and cnn-inferred depth. In: 2018 IEEE International Conference on Multimedia and Expo (ICME). pp. 1–6 (2018)

11. Klein, G., Murray, D.: Parallel tracking and mapping for small ar workspaces. In: 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality. pp. 225–234 (2007)
12. Liu, H., Zhang, G., Bao, H.: Robust keyframe-based monocular slam for augmented reality. In: 2016 IEEE International Symposium on Mixed and Augmented Reality (ISMAR). pp. 1–10 (2016)
13. Mur-Artal, R., Montiel, J.M.M., Tardós, J.D.: Orb-slam: A versatile and accurate monocular slam system. IEEE Transactions on Robotics **31**(5), 1147–1163 (2015)
14. Mur-Artal, R., Tardós, J.D.: Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. IEEE Transactions on Robotics **33**(5), 1255–1262 (2017)
15. Newcombe, R.A., Izadi, S., Hilliges, O., Molyneaux, D., Kim, D., Davison, A.J., Kohi, P., Shotton, J., Hodges, S., Fitzgibbon, A.: Kinectfusion: Real-time dense surface mapping and tracking. In: 2011 10th IEEE International Symposium on Mixed and Augmented Reality. pp. 127–136 (2011)
16. Newcombe, R.A., Lovegrove, S.J., Davison, A.J.: Dtam: Dense tracking and mapping in real-time. In: 2011 International Conference on Computer Vision. pp. 2320–2327 (2011)
17. Pan, Y., Xu, X., Ding, X., Huang, S., Wang, Y., Xiong, R.: Gem: Online globally consistent dense elevation mapping for unstructured terrain. IEEE Transactions on Instrumentation and Measurement **70**, 1–13 (2021)
18. Tateno, K., Tombari, F., Laina, I., Navab, N.: Cnn-slam: Real-time dense monocular slam with learned depth prediction. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
19. Teixeira, L., Chli, M.: Real-time local 3d reconstruction for aerial inspection using superpixel expansion. In: 2017 IEEE International Conference on Robotics and Automation (ICRA). pp. 4560–4567 (2017)
20. Wang, K., Ding, W., Shen, S.: Quadtree-accelerated real-time monocular dense mapping. In: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 1–9 (2018)
21. Wang, K., Gao, F., Shen, S.: Real-time scalable dense surfel mapping. In: 2019 International Conference on Robotics and Automation (ICRA). pp. 6919–6925 (2019)
22. Xue, T., Luo, H., Cheng, D., Yuan, Z., Yang, X.: Real-time monocular dense mapping for augmented reality. In: Proceedings of the 25th ACM International Conference on Multimedia. p. 510–518. MM '17, Association for Computing Machinery, New York, NY, USA (2017)
23. Yin, X., Wang, X., Du, X., Chen, Q.: Scale recovery for monocular visual odometry using depth estimated with deep convolutional neural fields. In: 2017 IEEE International Conference on Computer Vision (ICCV). pp. 5871–5879 (2017)
24. Younes, G., Asmar, D.C., Shammas, E.A.: A survey on non-filter-based monocular visual SLAM systems. CoRR **abs/1607.00470** (2016)