2021 IEEE International Conference on Robotics and Automation (ICRA 2021)
May 31 - June 4, 2021, Xi'an, China

Semantically Guided Multi-View Stereo for Dense 3D Road Mapping

Mingzhe Lv[1], Diantao Tu[1], Xincheng Tang[2], Yuqian Liu[3] and Shuhan Shen[1*]

Abstract— **Compared to widely used LiDAR-based mapping in autonomous driving field, image-based mapping method has the advantages of low cost, high resolution, and no need for complex calibration. However, the image-based 3D mapping depends heavily on the texture richness and always leaves holes and outliers in low-textured areas, such as the road surface. To this end, this paper proposed a novel semantically guided Multi-View Stereo method for dense 3D road mapping, which integrates semantic information into PatchMatch-based MVS pipeline and uses image semantic segmentation as soft constraints in neighbor views selection, depth-map initialization, depth propagation, and depth-map completion. Experimental results on public and our own datasets show that, with the help of semantics, the proposed method achieves superior completeness with comparable accuracy for 3D road mapping compared to state-of-the-art MVS methods.**

I. Introduction

Dense semantic 3D mapping of the road environment is a key requirement of autonomous vehicles. Once the dense point cloud is generated, a series of high-level perception tasks of autonomous driving could be carried out, such as 3D scene understanding, vectorized map construction, path planning, vehicle localization, etc. Generally, LiDAR is the common choice to generate dense and accurate point clouds and a batch of LiDAR-based mapping methods have been proposed in recent years [1]–[3]. In contrast, with the fast developments of Structure-from-Motion (SfM), Multi-View Stereo (MVS), and image semantic segmentation, the image-based dense semantic 3D mapping interests more and more researchers and autonomous driving companies. Compared to LiDAR mapping which suffers from low vertical resolution or LiDAR-image fused mapping which relies on complex LiDAR-camera calibration, a pure image-based 3D mapping system has the advantages of low cost, high resolution, no need for complex calibration, and is especially suitable for crowdsourced map collection. Besides, the 2D semantics and 3D geometries are more naturally integrated into the image-based 3D reconstruction pipeline.

Image-based 3D mapping pipeline usually contains three key steps. Firstly, the SfM algorithm is used to recover camera poses and sparse point cloud, then the MVS algorithm is applied to compute dense 3D points, and finally, 2D semantic segmentation results are used to label the category of each 3D point to form the dense semantic 3D map. In this paper, we focus on the middle step and propose a unified semantically guided MVS method to incorporate the semantic information into dense point cloud reconstruction process.

Among the various geometry and end-to-end MVS methods, PatchMatch-based MVS [4]–[7] achieve state-of-the-art performance in most popular MVS benchmarks [8] [9]. However, in autonomous driving field, the dense 3D road mapping typically focuses on traffic facilities, such as roads, traffic signs, traffic lights, poles, etc. Mapping of these facilities in wide open environment is challenging to the existing MVS algorithms which are more suited for well-textured scenes and ideally captured images.

Intuitively, image semantics could provide rich high-level information for the dense matching process to produce a more accurate and complete 3D representation. Several works have explored this idea [10]–[12], but they focus on building facades or indoor scenes, and there is still a challenging problem for the road scene reconstruction because of its wide open low-textured area.

In order to tackle these challenging problems, this paper proposed a novel semantically guided MVS method specially designed for the 3D road mapping scenario. In the proposed method, pixel-wise semantics are used as soft constraints for neighbor views selection, depth map initialization, depth propagation, and depth map completion processes. Experimental results on public and our own datasets show the proposed method could generate more complete dense point clouds with comparable accuracy compared to state-of-the-art methods. In summary, our main contributions are:

1) We propose a practical semantically guided MVS method for road scenes in which the semantic information is used to enhance the computing process of all steps in a PatchMatch-based MVS pipeline.

2) The proposed method could greatly improve the completeness while maintaining the accuracy of the final 3D map on real-world datasets compared with SOTAs.

II. Related Work

A lot of works are related to the dense 3D road mapping problem, and one can refer [13] for a comprehensive review. Here we focus on related works in two areas closely related to this paper, namely PatchMatch-based MVS and semantics and geometry fused MVS.

PatchMatch-based MVS. Multi-View Stereo (MVS) aims at reconstructing a dense 3D representation of the scene given a set of calibrated images. According to [14], 3D scenes could be represented in several ways, such as voxels, polygon meshes, depth maps, etc. And MVS algorithms

This work was supported by the National Natural Science Foundation of China (No. 61873265 and 61632003). *Corresponding author.

[1]National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China, and School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100049, China, and CASIA-SenseTime Research Group.

[2]School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing 100083, China

[3]SenseTime Research, Hangzhou 311215, China

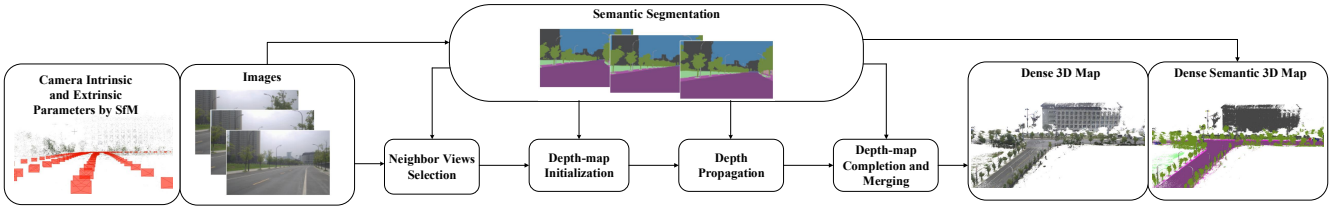

Fig. 1: The pipeline of proposed method. The inputs are images and their semantic segmentation results, as well as their intrinsic and extrinsic calibration parameters generated by SfM. The semantic information is integrated into each step of our method. The output is a dense 3D map of the road scene with optional semantics.

could be divided into four classes, called voxel-based methods [15], [16], surface evolution based methods [17]–[19], feature point growing based methods [20]–[22], and depth-map merging based methods [4], [5], [23]–[26]. Among these classes, the depth-map merging based methods have been proved to be more adapted to large-scale scenes, and PatchMatch-based MVS is the most representative one.

The idea of PatchMatch was first proposed by Barnes et al. [27] to match pixels between two images. Bleyer et al. [28] extended the 2D matching relationship like translation and scale to 3D, and Shen [4] further extended PatchMatch to Multi-View Stereo. Zheng et al. [29] jointly performed depth estimation and pixel-wise view selection by formulating them into a hidden Markov chain. Schnberger et al. [5] furthered this work which could jointly estimate the depth and normal, and perform pixel-wise view selection. Galliani et al. [30] modified the propagation scheme of PatchMatch so that it can be massively parallelized on GPU. Xu et al. [6] utilized downsampled images and median filter to estimate the coarse depth values and employed geometric consistency to guide the propagation of depth hypotheses to the higher resolution. Recent years, deep learning-based MVS methods have begun to appear and achieved remarkable progress [31]–[35], but PatchMatch-based method is still among the top performing approaches on popular MVS benchmarks [8] [9].

Semantics and geometry fused MVS. Since the quality of geometry-based MVS heavily depends on texture richness, using image semantics to boost its performance becomes a feasible way. Ladicky et al. [36] firstly use semantic labels and 3D geometry jointly with traditional energy-based methods to estimate depth maps. Subsequently, [10], [11], [37]–[40] use volumetric representations to jointly infer 3D shapes and semantic classes in a principled manner. Departing from the voxel-based semantic methods, [41], [42] use semantic labels to impose class-specific shape knowledge in mesh refinement approaches. The common ground of these methods is that they use local shapes to influence the appearance-based class labels and vice versa.

More recently, Stathopoulou et al. [12] integrate semantic priors into PatchMatch-based MVS which has some similar ideas to ours, but they focus on plane fitting and cost function in depth map computation process, while we leverage semantics to boost all the steps in PatchMatch-based MVS. Besides, their application scenarios mainly concentre on build facades and indoor scenes, while ours are on road mapping.

III. Semantically Guided MVS

The inputs of our method are calibrated images and their semantic segmentation results. The intrinsic (focal length, principle point, distortions) and extrinsic (camera 6-DOF poses) calibration parameters could be generated using off-the-shelf SfM systems [43] [44]. The pipeline of the proposed method is shown in Fig. 1, which consists of four steps under the guidance of semantic information. We detail each step in the following subsections.

A. Semantically Guided Neighbor Views Selection

Neighbor views selection is a key step in the PatchMatch-based MVS, which aims to select several neighbor images for each reference image for the following depth map computation process. Based on the SfM result, i.e. camera poses, sparse 3D points with visibilities, we tend to select neighbor images that have sufficient co-visible sparse points in region of interest (road, traffic sign, etc.) for road mapping and sufficient ray intersection angles with the reference image.

First, we need to define a score S to measure the quality of a selected neighbor image. Suppose I_{ref} and I_{src} is a reference and a neighbor image respectively, and X denotes a 3D SfM point in the world coordinate which can be seen by I_{ref} and I_{src}. V_1 and V_2 are two vectors from the center of I_{ref} and I_{src} to X. By projecting point X to I_{ref} and I_{src}, we can calculate the projection depth d_1 and d_2. Then the score S between I_{ref} and I_{src} is computed as:

$$S = \sum_{X \in I_{ref} \cap I_{src}} \omega_a(X) \cdot \omega_d(X) \cdot \omega_s(X) \tag{1}$$

where $\omega_a(X) = \min((\theta/\alpha)^{1.5}, 1)$ is the angle weight, θ is the angle between V_1 and V_2, and α is an angle threshold in degree ($\alpha = 10$ in this paper). $\omega_d(X)$ is the distance ratio weight with $r = d_1/d_2$, defined as:

$$\omega_d(X) = \begin{cases} r^2 & r < 1/\beta \\ 1 & 1/\beta \le r \le \beta \\ \left(\frac{\beta}{r}\right)^2 & r > \beta \end{cases} \tag{2}$$

where β is a distance ratio threshold ($\beta = 1.6$ in this paper). Finally, $\omega_s(X)$ is a semantic weight, defined as:

$$\omega_s(X) = \begin{cases} 1.0 & L_X : \text{road facilities} \\ 0 & L_X : \text{dynamic objects} \\ 0.2 & L_X : \text{others} \end{cases} \tag{3}$$

where L_X refers to the semantic label of X's projection pixel on I_{ref} according to the categories of Cityscapes [45]. In this paper, we set high weights ($\omega_s(X) = 1$) for road facilities, such as roads and traffic signs, set low weights ($\omega_s(X) = 0.2$) for other parts, such as trees, vegetations and buildings, and set zero weights ($\omega_s(X) = 0$) for the dynamic objects like pedestrians, bikes, or vehicles.

After computing the score of each co-visible sparse point X between I_{ref} and I_{src}, these scores will be added together as the final score S. The above angle weight, distance ratio weight, and semantic weight tend to select neighbor images that have a large enough viewing angle (wide baseline) with the reference image, have a similar scene distance to the reference image, and have more co-visible points on road facilities. For each reference image I_{ref}, co-visible images with the highest N scores are selected as its neighbor views ($N = 10$ in this paper).

Note that due to the weak texture nature of road surfaces, there are usually few feature points in the road area, resulting in fewer SfM points, and may lead to inappropriate selection of neighbor images for road area reconstruction. Thus, compared to traditional neighbor views selection methods [21], [26], [46], the proposed semantic weighted selection method could effectively improve the reconstruction completeness of road areas, which is demonstrated in the experiments part.

B. Semantically Guided Depth Map Initialization

In the PatchMatch-based MVS, a support plane $\{d,n\}$ is given for each pixel in the image, which represents the local tangent plane of scene surface. Here d and n is the depth and normal of the plane respectively. The conventional PatchMatch [4], [5] use a random initialization method to generate initial $\{d,n\}$, i.e. a random d in the depth range defined by all visible SfM points for this image and a random n within the visible range of the image. The core idea of the above random initialization process is that it is very likely to have at least one good guess for each scene plane in the image, especially for high resolution images in which each scene plane contains plenty of pixels.

Recently, some methods try to give more appropriate initial values than pure random. Based on the prior depth of SfM points, [47] refers to the depth of four nearest SfM points for each pixel to reduce the minimum and maximum random range. [48] uses 2D Delaunay triangulation and its back projection to generate initial depth values. Based on these ideas and by further taking into account the semantic information, we propose a semantically guided initialization method to generate more accurate initial depth and normal for each pixel in road mapping.

For each reference image I_{ref}, we project all visible SfM points onto I_{ref} to get their corresponding 2D reprojection points with semantics, and then perform 2D Delaunay triangulation on these points to get a 2D mesh on I_{ref}. Then a 3D mesh corresponding to this 2D mesh could be obtained directly because each vertex on this 2D mesh is a reprojection of a 3D SfM point and the location of the 3D point is known. That is, the 2D mesh on I_{ref} is the reprojection of the 3D mesh whose vertices are visible 3D SfM points of I_{ref}.

Intuitively, for each facet f in the 2D mesh, if its three vertices have the same semantic labels, all pixels in f are likely on the same scene plane, otherwise, there may be a depth discontinuity in this facet. So for each pixel in f, its initial depth refers to the vertex of f with the same semantics.

More specifically, for each pixel p in the facet f with three same semantic vertices, we construct a viewing ray from the camera center to p, and find the intersection point of this ray and its corresponding 3D facet in space. In this way, the initial depth d of p is set as the reprojection depth of this intersection point, and the initial normal n is set as the normal of the 3D facet.

For each pixel p in the facet f with different semantic vertices, if the semantic label of p is different from all three vertices, the depth d of p is randomly selected within the depth range of f, and if the semantic label of p is the same as one or two vertices, the depth of p is set as the mean depth of the same semantic vertices plus a random disturbance in the depth range of f. Besides, the initial normal n of p is set as the normal of the 3D facet plus a small random disturbance (e.g. 10 degrees).

C. Scale-Adaptive Depth Propagation

After the initialization, each pixel in the image is associated with a support plane $\{d,n\}$. Then the spatial propagation process is carried out to refine the depth and normal of pixels. The spatial propagation is the core idea in the PatchMatch-based MVS, which is used to propagate good guess of support plane $\{d,n\}$ to neighbor pixels. The propagation starts from the upper left corner of the reference image and moves towards the lower right corner. For each pixel, We set a square window centered on this pixel, and check its photometric consistency, such as Normalized Cross Correlation (NCC), with its corresponding homography projection window on the neighbor image. This process is repeated until it reaches the bottom right corner. Then we reverse the propagation order to visit the pixels from the bottom-right to the top-left and iterates 3-5 times until the end.

During the depth spatial propagation process, the NCC window size is critical for the photometric consistency measurement. A smaller window is difficult to handle weak texture areas, and a larger window may over-smooth the depth discontinuous area. Though some methods use pyramid images [6] or texture-masks [47] to set a more reasonable window size, their ability to handle weak textures is still relatively limited, especially when there is a large continuous weakly textured area. To this end, in this section we proposed a scale-adaptive depth spatial propagation method, in which the texture richness is measured for each pixel and the window size is given accordingly.

For each pixel p in the reference image, we set a square window B of size $l_{min} \times l_{min}$ ($l_{min} = 5$ pixels in this paper) centered on p, and use the variance of the intensity values of the pixels in the window to measure its texture richness, as $s(B) = \frac{1}{|B|}\sum_{x_i \in B}(x_i - \bar{x})^2$, in which x_i is the intensity of pixel

in B, $\bar{x}$ is the mean intensity in B, and $|B|$ is the number of pixels in B. Obviously, the larger $s(B)$ is, the richer of texture in this window. For convenience, we further define a normalized texture richness $t(B) = \frac{1}{1+s(B)}$, $t(B) \in (0,1]$. The smaller the $t(B)$, the richer the texture, and vice versa. Thus, according to this normalized texture richness measurement, the matching window B used for NCC is enlarged to $l_{enl} \times l_{enl}$, and l_{enl} is computed as:

$$l_{enl} = l_{min} + \lceil \frac{l_{max} - l_{min}}{1 + e^{-10(t(B)-0.5)}} \rceil \tag{4}$$

where, l_{max} is the upper size of the window (e.g. $l_{max} =$ 37 pixels), and $\lceil * \rceil$ represents the ceiling function. Eq.4 indicates that the side length of the window B will increase by a sigmod function from l_{min} to l_{max} as $t(B)$ increases.

Given the enlarged window B=$l_{enl} \times l_{enl}$ for each pixel p, as well as the local plane $\{d,n\}$ stored for p, its corresponding window pixels in the neighbor image are computed by Plane-induced Homography mapping [49] between the two images. Here, we use bilateral weighted normalized NCC to measure the photometric consistency between two patches, as:

$$\text{NCC} = \frac{cov_\omega(B, H(B))}{\sqrt{cov_\omega(B,B) cov_\omega(H(B), H(B))}} \tag{5}$$

where $H(B)$ is the Homography mapping of B on the neighbor image. $cov_\omega(X,Y) = E_\omega(X - E(X))E_\omega(Y - E(Y))$ is the weighted covariance and $E_\omega(X) = \sum_i \omega_i x_i / \sum_i \omega_i$ is the weighted average. The weight ω_i is computed as:

$$\omega_i = \exp(-\frac{\Delta g_i^2}{2\sigma_g^2} - \frac{\Delta d_i^2}{2\sigma_d^2} - \frac{\Delta L_i^2}{2\sigma_L^2}) \tag{6}$$

where Δg_i=$|x_i - x_p|$ is the intensity distance in which x_i is the intensity of pixel i in the matching window, Δd_i=$\|\mathbf{u}_i - \mathbf{u}_p\|$ is the spatial distance in which $\mathbf{u}_i$ is the image coordinate of pixel i, ΔL_i=Bool$(L_i - L_p)$ is the semantic difference in which L_i is the semantic label of pixel i, and ΔL_i=0 if L_i=L_p, otherwise ΔL_i=1. The importance of the three weights is scaled by factor σ_g, σ_d and σ_L, which is respectively set to 0.2, $l_{enl}/2$ and 0.1 in this paper. The weight ω_i measures the difference in intensity, distance, and semantics at the same time to indicate the likelihood that a pixel i in the matching window B belongs to the same plane as its center pixel p.

Since a large matching window may cause the computation of NCC is rather slow, we adopt two strategies to speed up. First, we skip all pixels that are labeled as *sky* because they are useless for road mapping and the segmentation results of *sky* are generally very accurate. Second, we adopt a discrete sampling method in window B, that is, we increase the sampling step as the window size increases to ensure that the number of pixels involved in NCC computation of different sized window is basically the same.

D. Semantically Guided Depth Completion and Merging

Since the raw depth maps may not completely agree with each other due to depth errors, a filtering process is carried out to enforce geometric consistency over neighbor views. For each pixel in the reference image, we back project it to 3D and then project it to its neighbor views. If the projected depth d_R is consistent with the depth d_N in the neighbor depth map $(d_R - d_N)/d_N < \tau$ ($\tau = 0.01$ in this paper) on at least two neighbor views, the point is regarded as stable. Otherwise, it will be removed from the depth map.

The depth map filtering process could remove lots of noisy depths, but may also leave holes in some areas. Therefore, we perform a semantically guided completion process to further improve the completeness of the depth map, especially for the area of road facilities. In the completion step, we first compute 2D Delaunay triangulation on the filtered depth map. For those areas that have obvious planar properties, such as road surface and traffic signs, the empty pixels could be completed by viewing ray interaction. More specifically, if the three vertexes of a facet have the same semantic label (road or traffic sign), all empty pixels that have same semantics to vertexes within this facet could get their depths by viewing ray intersection with its corresponding 3D facet in space as described in the depth initialization part (Sec.III.B). But in two cases the completion step will be skipped, one is that the facet is too large (side length exceeds 100 pixels in this paper), another is that depth difference between the three vertices of the facet is too large ($\pm$5% of the mean depth), because in both cases the facet may contain other categories of objects or non-planar regions.

Finally, we back project all depth maps into 3D and merge them into a point cloud, and the semantic label of each point is given by max-voting using the semantics from all its visible images. According to the needs of road map generation, points with semantic labels as dynamic objects will be removed, such as vehicles and pedestrians.

IV. Experiments

A. Datasets

To demonstrate the effectiveness of the proposed method, we use both public and our own datasets for quantitative and qualitative evaluation. The first group of datasets is from the KITTI odometry benchmark [50], which contains 22 stereo image sequences and synchronized LiDAR data, and we use 3 of them for our evaluation, named Kitti-01 (2202 images, 2.4km) Kitti-04 (542 images, 0.4km), and Kitti-06 (2202 images, 1.0km). These three sequences include different road scenarios, such as beltway, urban roads, and factory-in roads. For each dataset, we extract and match SIFT features between images, triangulate feature tracks, and perform global Bundle Adjustment using the GNSS/IMU poses provided by the dataset as initialization to refine intrinsic and extrinsic camera parameters. Then we merge the LiDAR points using camera poses from the SfM result, and use the merged LiDAR points as ground-truth to evaluate the MVS road mapping result.

The second group of datasets, named Urban-01 (1299 images, 1.2km) and Urban-02 (8428 images, 3.1km), are collected using our own road mapping vehicle, which has similar hardware configuration as KITTI. Compared to KITTI, the Urban datasets have much more images with higher image resolution (1920$\times$1200), and contain much larger and

more weakly textured road areas which are more challenging for the image-based road mapping.

Since the LiDAR map may contain dynamic objects, such as moving vehicles and pedestrians, which will cause errors in quantitative evaluation, we manually selected 7 sections without dynamic objects from the LiDAR maps, including two sections from Kitti-01 (Kitti-01-Sec1, Kitti-01-Sec2), one section from Kitti-04 (Kitti-04-Sec), one section from Kitti-06 (Kitti-06-Sec), one section from Urban-01 (Urban-01-Sec), and two sections from Urban-02 (Urban-02-Sec1, Urban-02-Sec2), for quantitative evaluation. Some snapshots of the datasets are shown in Fig.2.

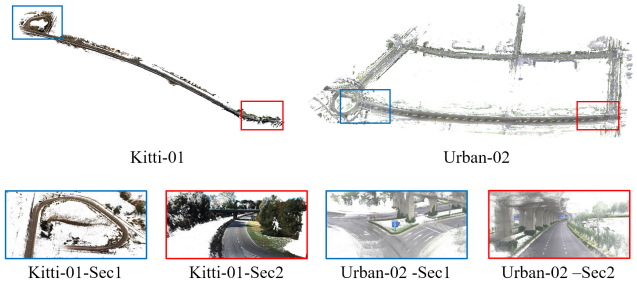

Fig. 2: Snapshots of Kitti-01 and Urban-02, and their local sections used for quantitative evaluation.

For all the datasets, we use DeepLab V3+ [51] pre-trained on Cityscape [45] to generate the semantic segmentation results in all our experiments. Note that our method has a certain robustness to the semantic segmentation quality, because semantics are used as soft constraints in each step of our pipeline. Therefore, we didn't fine-tune the network on our datasets, but directly use the pre-trained model for segmentation.

B. Individual Step and Ablation Study

To investigate the influence of semantic information on each module, in this section we conduct individual step experiment and ablation study. First, the result of each step in our pipeline is reported.

Neighbor views selection: The neighbor views selection result of a reference image near a crossroad is shown in Fig.3. In this case, visible points of this reference image are dominated by vegetation and bridge piers, thus the neighbor images selected by traditional method [46] is not conducive to the reconstruction of road areas. In contrast, with the help of semantics, the neighbor views selected by our method are more reasonable for road mapping.

Depth map initialization: The depth map initialized using random initialization, triangulated mesh initialization, and our semantically guided initialization is shown in Fig.4. The results show that our initialization method could generate more reasonable initial depth values, thus can improve the accuracy of refined depth map after depth propagation.

Depth map propagation: As shown in Fig.5 (first three rows), the traditional propagation method with a small window may leave lots of holes in the depth map due to weak texture, and a large window will over-smooth the depth boundary. With the help of our scale-adaptive window and semantics, the depth map obtained by our method is more complete with clear edges.

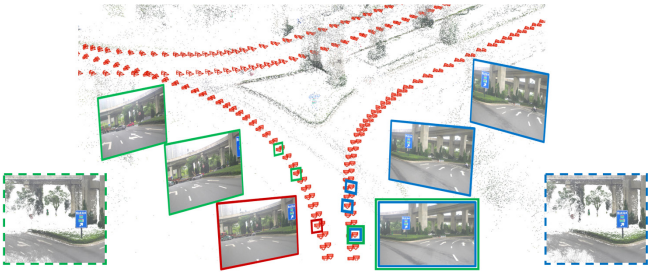

Fig. 3: Neighbor views selection results. The reference image is shown in red rectangle, selected neighbor views w/ and w/o semantic information are shown in green and blue rectangles respectively, and the final 3D map in this place is shown in green and blue dashed rectangles respectively.

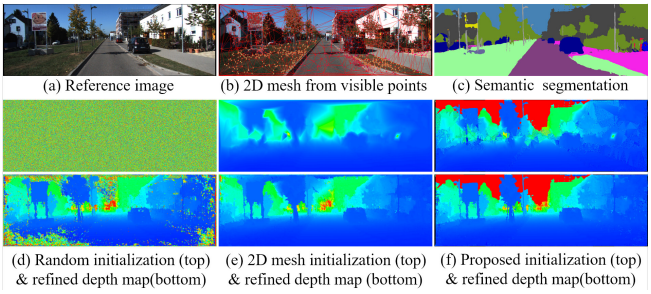

Fig. 4: Depth map initialization results

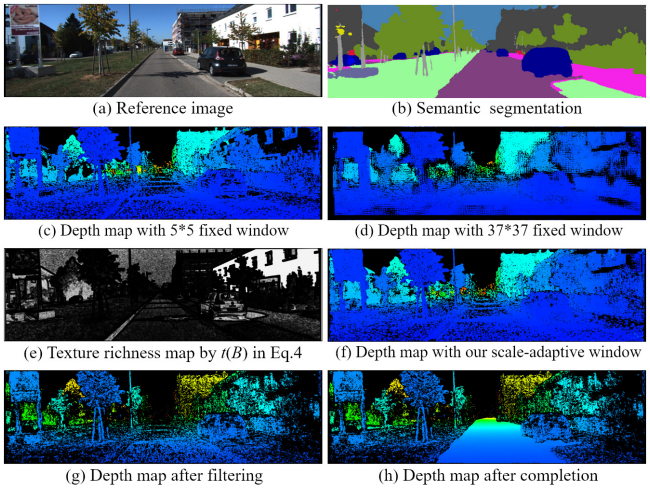

Fig. 5: Depth map propagation and completion results

Depth map completion: The depth map filtering and completion are crucial for removing outliers and preserving completeness, and Fig.5 (bottom row) shows an example. The results show that the proposed semantically guided completion process could effectively complete the road surface as shown in Fig.5(h).

In addition to the qualitative comparison of each individual step, we also conducted quantitative ablation experiments on

TABLE I: Quantitative evaluation results of different methods on datasets (F-score/Precision/Recall).

	Kitti-01-Sec1	Kitti-01-Sec2	Kitti-04-Sec	Kitti-06-Sec	Urban-01-Sec	Urban-02-Sec1	Urban-02-Sec2
OpenMVS [48]	14.5/88.9/7.94	39.9/**97.4**/25.1	28.5/99.1/16.6	74.1/91.0/62.5	20.3/92.9/11.4	44.3/94.6/28.9	25.5/**89.8**/14.8
CasMVSNet [34]	49.0/68.4/38.1	65.6/95.3/50.0	86.2/97.2/77.5	83.4/89.1/78.4	52.3/76.0/39.8	0.05/3.75/0.02	19.5/63.2/11.5
VisMVSNet [35]	74.9/87.4/65.5	88.0/96.7/80.6	81.3/99.2/68.8	86.7/93.1/81.2	77.8/91.6/67.6	44.0/91.3/29.0	52.1/84.8/37.6
Ours	**85.9/89.4/82.7**	**96.1**/97.3/**94.9**	**97.3/99.3/95.4**	**91.8/95.0/88.7**	**87.9/94.2/82.3**	**88.8/96.9/81.9**	**86.5**/81.7/**91.9**

Urban-02-Sec1, and the result is shown in Table II. Here, we use the evaluation indicators introduced in Tanks and Temples benchmark [9], namely precision, recall, and F-score. The inlier threshold for MVS-LiDAR point correspondence is set to 10cm in this paper, i.e. a MVS point is considered as an inlier if the distance to its nearest LiDAR map point is smaller than 10cm. Table II shows that compared with the baseline PatchMatch-based MVS without semantic information, the proposed semantically guided MVS method achieves a significant improvement in the recall rate while maintaining precision, which indicates the completeness of the 3D map is largely improved. And the improvement of completeness could be seen clearly in Fig.6.

TABLE II: Ablation study on Urban-02-Sec1. SegView: use semantics for neighbor views selection, SegInit: use semantics in depth map initialization, SegProp: use scale-adaptive window and semantics in depth propagation, SegComp: use semantics in depth map completion.

Method	F-score/Precision/Recall
w/o SegView+SegInit+SegProp+SegComp	31.0/94.9/18.5
w/o SegView	71.3/96.5/56.5
w/o SegInit	85.1/97.2/75.7
w/o SegProp	86.2/96.8/77.7
w/o SegComp	77.6/**97.3**/64.6
w/ SegView+SegInit+SegProp+SegComp	**88.8**/96.9/**81.9**

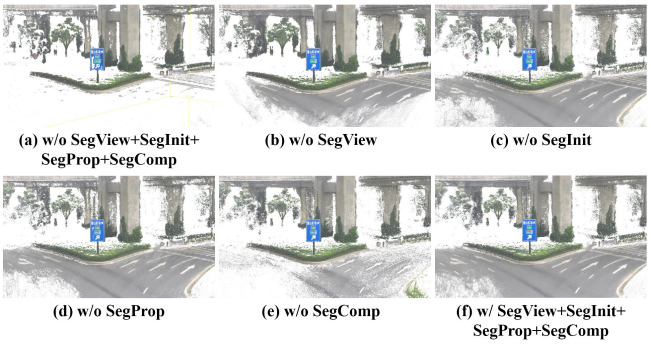

Fig. 6: Qualitative comparison of merged point cloud w/ and w/o semantics on Urban-02-Sec1. The meaning of SegView, SegInit, SegProp and SegComp is shown in the caption of Table II.

C. Comparison with SOTA

In this section, we compared our method with OpenMVS [48] which is the baseline open-sourced PatchMatch-based MVS system and among the top performing methods on popular MVS benchmarks. Besides, two best-performing learning-based MVS methods, CasMVSNet [34] and VisMVSNet [35] are also used for comparison. All methods are quantitative and qualitative evaluated on our datasets, shown in Table I and Fig.7. The results show that our method achieved better results on all datasets, especially on Urban-01 and Urban-02 with weak road textures.

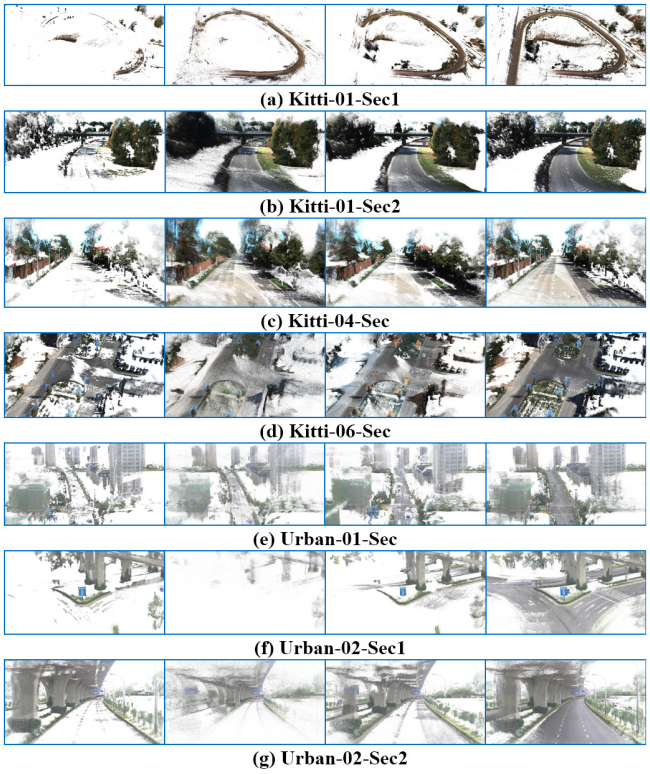

Fig. 7: The final 3D map generated by different methods on the datasets. In (a)-(g), from left to right, OpenMVS [48], CasMVSNet [34], VisMVSNet [35] and ours.

V. Conclusions

Compared to widely used LiDAR-based 3D road mapping, image-based mapping has the advantages of low cost and high resolution, but the completeness of the 3D map obtained by existing MVS methods is hard to guarantee. To this end, this paper proposed a semantically guided MVS method in which the image semantics is integrated into the PatchMatch-based MVS pipeline. Experimental results show that, with the help of semantics, the proposed method could generate more complete dense 3D road maps with comparable accuracy compared to state-of-the-art MVS methods.

REFERENCES

[1] R. W. Wolcott and R. M. Eustice, "Visual localization within lidar maps for automated urban driving," in *IROS*, 2014.

[2] K. Yoneda, H. T. Niknejad, T. Ogawa, N. Hukuyama, and S. Mita, "Lidar scan feature for localization with highly precise 3-d map," in *IROS*, 2014.

[3] R. W. Wolcott and R. M. Eustice, "Robust lidar localization using multiresolution gaussian mixture maps for autonomous driving," *The International Journal of Robotics Research*, vol. 36, no. 3, pp. 292–319, 2017.

[4] S. Shen, "Accurate multiple view 3d reconstruction using patch-based stereo for large-scale scenes," *IEEE transactions on image processing*, vol. 22, no. 5, pp. 1901–1914, 2013.

[5] J. L. Schönberger, E. Zheng, J.-M. Frahm, and M. Pollefeys, "Pixelwise view selection for unstructured multi-view stereo," in *ECCV*. Springer, 2016.

[6] Q. Xu and W. Tao, "Multi-scale geometric consistency guided multi-view stereo," in *CVPR*, 2019.

[7] A. Romanoni and M. Matteucci, "Tapa-mvs: Textureless-aware patchmatch multi-view stereo," in *ICCV*, 2019.

[8] T. Schöps, J. L. Schönberger, S. Galliani, T. Sattler, K. Schindler, M. Pollefeys, and A. Geiger, "A multi-view stereo benchmark with high-resolution images and multi-camera videos," in *CVPR*, 2017.

[9] A. Knapitsch, J. Park, Q.-Y. Zhou, and V. Koltun, "Tanks and temples: Benchmarking large-scale scene reconstruction," *ACM Transactions on Graphics (ToG)*, vol. 36, no. 4, pp. 1–13, 2017.

[10] C. Häne, C. Zach, A. Cohen, and M. Pollefeys, "Dense semantic 3d reconstruction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, pp. 1730–1743, 2016.

[11] N. Savinov, C. Häne, L. Ladicky, and M. Pollefeys, "Semantic 3d reconstruction with continuous regularization and ray potentials using a visibility consistency constraint," in *CVPR*, 2016.

[12] E. K. Stathopoulou, R. Battisti, D. Cernea, F. Remondino, and A. Georgopoulos, "Semantically derived geometric constraints for mvs reconstruction of textureless areas," *Remote Sensing*, vol. 13, no. 6, 2021.

[13] J. Janai, F. Gney, A. Behl, and A. Geiger, *Computer Vision for Autonomous Vehicles: Problems, Datasets and State of the Art*. Foundations and Trends in Computer Graphics and Vision, 2020.

[14] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski, "A comparison and evaluation of multi-view stereo reconstruction algorithms," in *CVPR*, 2006.

[15] G. Vogiatzis, C. Hernandez, P. H. Torr, and R. Cipolla, "Multi-view stereo via volumetric graph-cuts and occlusion robust photo-consistency," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 12, pp. 2241–2246, 2007.

[16] S. N. Sinha, P. Mordohai, and M. Pollefeys, "Multi-view stereo via graph cuts on the dual of an adaptive tetrahedral mesh," in *ICCV*, 2006.

[17] C. Hernandez and F. Schmitt, "Silhouette and stereo fusion for 3d object modeling," *Computer Vision and Image Understanding*, vol. 96, no. 3, pp. 367–392, 2004.

[18] V. H. Hiep, R. Keriven, P. Labatut, and J.-P. Pons, "Towards high-resolution large-scale multi-view stereo," in *CVPR*, 2009.

[19] D. Cremers and K. Kolev, "Multiview stereo and silhouette consistency via convex functionals over convex domains," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 6, pp. 1161–1174, 2011.

[20] M. Lhuillier and L. Quan, "A quasi-dense approach to surface reconstruction from uncalibrated images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 3, pp. 418–433, 2005.

[21] M. Goesele, N. Snavely, B. Curless, H. Hoppe, and S. M. Seitz, "Multi-view stereo for community photo collections," in *ICCV*, 2007.

[22] Y. Furukawa and J. Ponce, "Accurate, dense, and robust multiview stereopsis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 8, pp. 1362–1376, 2010.

[23] M. Goesele, B. Curless, and S. M. Seitz, "Multi-view stereo revisited," in *CVPR*, 2006.

[24] N. D. Campbell, G. Vogiatzis, C. Hernandez, and R. Cipolla, "Using multiple hypotheses to improve depth-maps for multi-view stereo," in *ECCV*, 2008.

[25] E. Tola, C. Strecha, and P. Fua, "Efficient large-scale multi-view stereo for ultra high-resolution image sets," *Machine Vision and Applications*, vol. 23, no. 5, pp. 903–920, 2012.

[26] C. Bailer, M. Finckh, and H. P. Lensch, "Scale robust multi view stereo," in *ECCV*, 2012.

[27] C. Barnes, E. Shechtman, A. Finkelstein, and D. B. Goldman, "Patchmatch: A randomized correspondence algorithm for structural image editing," in *ACM Transactions on Graphics*, vol. 28, no. 3. ACM, 2009, p. 24.

[28] M. Bleyer, C. Rhemann, and C. Rother, "Patchmatch stereo-stereo matching with slanted support windows." in *BMVC*, 2011.

[29] E. Zheng, E. Dunn, V. Jojic, and J.-M. Frahm, "Patchmatch based joint view selection and depthmap estimation," in *CVPR*, 2014.

[30] S. Galliani, K. Lasinger, and K. Schindler, "Massively parallel multi-view stereopsis by surface normal diffusion," in *ICCV*, 2015.

[31] P.-H. Huang, K. Matzen, J. Kopf, N. Ahuja, and J.-B. Huang, "Deepmvs: Learning multi-view stereopsis," in *CVPR*, 2018.

[32] Y. Yao, Z. Luo, S. Li, T. Fang, and L. Quan, "Mvsnet: Depth inference for unstructured multi-view stereo," in *ECCV*, 2018.

[33] Y. Yao, Z. Luo, S. Li, T. Shen, T. Fang, and L. Quan, "Recurrent mvsnet for high-resolution multi-view stereo depth inference," in *CVPR*, 2019.

[34] X. Gu, Z. Fan, S. Zhu, Z. Dai, F. Tan, and P. Tan, "Cascade cost volume for high-resolution multi-view stereo and stereo matching," in *CVPR*, 2020.

[35] J. Zhang, Y. Yao, S. Li, Z. Luo, and T. Fang, "Visibility-aware multi-view stereo network," in *BMVC*, 2020.

[36] L. Ladickỳ, P. Sturgess, C. Russell, S. Sengupta, Y. Bastanlar, W. Clocksin, and P. H. Torr, "Joint optimization for object class segmentation and dense stereo reconstruction," *International Journal of Computer Vision*, vol. 100, no. 2, pp. 122–133, 2012.

[37] C. Hane, C. Zach, A. Cohen, R. Angst, and M. Pollefeys, "Joint 3d scene reconstruction and class segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 97–104.

[38] M. Blaha, C. Vogel, A. Richard, J. D. Wegner, T. Pock, and K. Schindler, "Large-scale semantic 3d reconstruction: an adaptive multi-resolution model for multi-class volumetric labeling," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3176–3184.

[39] I. Cherabier, J. L. Schonberger, M. R. Oswald, M. Pollefeys, and A. Geiger, "Learning priors for semantic 3d reconstruction," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 314–330.

[40] A. O. Ulusoy, M. J. Black, and A. Geiger, "Semantic multi-view stereo: Jointly estimating objects and voxels," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017, pp. 4531–4540.

[41] M. Blaha, M. Rothermel, M. R. Oswald, T. Sattler, A. Richard, J. D. Wegner, M. Pollefeys, and K. Schindler, "Semantically informed multiview surface refinement," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 3819–3827.

[42] A. Romanoni, M. Ciccone, F. Visin, and M. Matteucci, "Multi-view stereo with single-view semantic mesh refinement," in *ICCV Workshops*, 2017.

[43] P. Moulon, P. Monasse, R. Perrot, and R. Marlet, "Openmvg: Open multiple view geometry," in *International Workshop on Reproducible Research in Pattern Recognition*. Springer, 2016.

[44] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," in *CVPR*, 2016.

[45] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *CVPR*, 2016, pp. 3213–3223.

[46] S. Shen and Z. Hu, "How to select good neighboring images in depth-map merging based 3d modeling," *IEEE Transactions on Image Processing*, vol. 23, no. 1, pp. 308–318, 2013.

[47] Z. Xu, Y. Liu, X. Shi, Y. Wang, and Y. Zheng, "Marmvs: Matching ambiguity reduced multiple view stereo for efficient large scale scene reconstruction," in *CVPR*, 2020.

[48] openMVS, https://github.com/cdcseacave/openMVS.

[49] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, ISBN: 0521540518, 2004.

[50] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *CVPR*, 2012.

[51] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *ECCV*, 2018.