Patch-Based Semantic Labeling of Road Scene Using Colorized Mobile LiDAR Point Clouds

Huan Luo, Cheng Wang, *Member, IEEE*, Chenglu Wen, *Member, IEEE*, Zhipeng Cai, Ziyi Chen, Hanyun Wang, *Member, IEEE*, Yongtao Yu, and Jonathan Li, *Senior Member, IEEE*

***Abstract*—Semantic labeling of road scenes using colorized mobile LiDAR point clouds is of great significance in a variety of applications, particularly intelligent transportation systems. However, many challenges, such as incompleteness of objects caused by occlusion, overlapping between neighboring objects, interclass local similarities, and computational burden brought by a huge number of points, make it an ongoing open research area. In this paper, we propose a novel patch-based framework for labeling road scenes of colorized mobile LiDAR point clouds. In the proposed framework, first, three-dimensional (3-D) patches extracted from point clouds are used to construct a 3-D patch-based match graph structure (3D-PMG), which transfers category labels from labeled to unlabeled point cloud road scenes efficiently. Then, to rectify the transferring errors caused by local patch similarities in different categories, contextual information among 3-D patches is exploited by combining 3D-PMG with Markov random fields. In the experiments, the proposed framework is validated on colorized mobile LiDAR point clouds acquired by the RIEGL VMX-450 mobile LiDAR system. Comparative experiments show the superior performance of the proposed framework for accurate semantic labeling of road scenes.**

***Index Terms*—Semantic labeling, 3D-PMG, Markov random field, colorized mobile LiDAR point clouds.**

I. Introduction

SEMANTIC labeling of road scenes automatically assigns a category label to each basic element (eg. pixel or point) in road scenes and has increasingly attracted the attention of researchers because of its importance in a variety of applications, such as urban planning, environmental impact assessment, cultural heritage documentation, etc. Particularly, in intelligent transportation systems (ITS), for example, semantic labeling of road scenes may assist Advanced Driver Assistance Systems (ADASs) in making correct decisions and responses by interpreting the surrounding environment of the host vehicle and understanding objects in the context of a traffic scene. Many researches regarding ITS have mainly focused on exploiting road-related information, including road structure [1]–[3], road curbs [4]–[6], and road markings [7]–[11]. In fact, the context information surrounding the road, such as the locations of light poles, trees, and vehicles, may also provide important cues for understanding complex road environments, even with little road-related context information. These surrounding context information can be obtained through road scene labeling.

In the past decades, semantic labeling based on 2D images has achieved great success [12]–[18]. However, in traditional optical imaging-based systems, the intrinsic deficiencies, including lack of accurate geospatial information, image distortions caused by camera lens, image qualities influenced by illumination conditions, and severe occlusions, limit the extensive use of on-image semantic labeling in ITS applications.

With the rapid development of light detection and ranging (LiDAR) technologies in recent years, the acquisition of large volumes of highly dense and accurate colorized point clouds with geospatial information over a large area is easily and rapidly accomplished by mobile LiDAR systems [19]. Compared with traditional optical imaging-based systems, mobile LiDAR systems can capture real-world coordinates of road scenes in a short time period. Mobile LiDAR systems are immune to impacts of illumination conditions, occlusions, and image distortions. Therefore, semantic labeling based on colorized mobile LiDAR point clouds (see Fig. 1) of road scenes can effectively alleviate the above problems of semantic labeling based on images. Moreover, compared to mobile LiDAR point clouds without color information, colorized mobile LiDAR point clouds provide not only geometric information but also texture information that is greatly essential to on-image semantic labeling. Semantic labeling using colorized mobile LiDAR point clouds can be an effective alternative to road scenes labeling.

In spite of these advantages of colorized mobile LiDAR point clouds, semantic labeling of road scenes based on colorized mobile LiDAR point cloud is still an open issue because of its challenges, including incompleteness of objects caused by occlusions, overlapping between neighboring objects, interclass local similarities, and computational burden brought by a huge number of points. In this paper, we propose a novel patch-based framework for accurate semantic labeling of colorized

Manuscript received May 5, 2015; revised August 31, 2015; accepted November 5, 2015. This work was supported by the National Natural Science Foundation of China under Grant 61401382 and Grant 61371144. The Associate Editor for this paper was S. S. Nedevschi. *(Corresponding author: Chenglu Wen.)*

H. Luo, C. Wang, C. Wen, Z. Cai, Z. Chen, and Y. Yu are with the Key Laboratory of Sensing and Computing for Smart Cities, Xiamen University, Xiamen 361005, China (e-mail: scholar.luo@gmail.com; cwang@xmu.edu.cn; clwen@xmu.edu.cn; azptc2h@gmail.com; chenziyicip@163.com; allennessy.yu@gmail.com).

H. Wang is with the School of Electronic Science and Engineering, National University of Defense Technology, Changsha 410073, China (e-mail: why.scholar@gmail.com).

J. Li is with the Key Laboratory of Underwater Acoustic Communication and Marine Information Technology of the Ministry of Education, Xiamen University, Xiamen 361005, China, and also with the Department of Geography and Environmental Management, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: junli@xmu.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2015.2499196

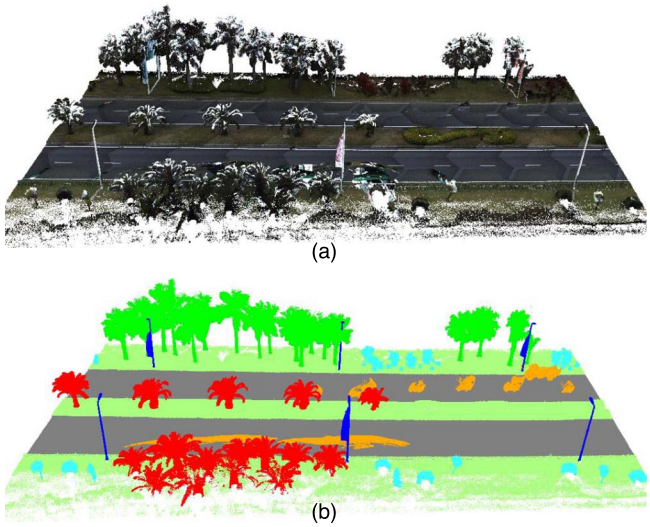


Fig. 1. Example of semantic labeling of road scene using colorized mobile LiDAR point clouds: (a) road scene of colorized mobile LiDAR point cloud; (b) semantic labeling results.

mobile LiDAR point clouds of road scenes. In the proposed framework, firstly, 3D local patches extracted from labeled point clouds are used to train a novel 3D patch-based match graph structure (3D-PMG). Then category labels are transferred from labeled to unlabeled point clouds by the trained 3D-PMG. Finally, in order to refine the incorrect category labels caused by local 3D patch similarity in different categories, a Markov Random Field (MRF) [20] model is integrated into the proposed framework to consider contextual information. The proposed framework can effectively handle the aforementioned challenges by introducing patch-based matches and contextual information among patches. Also, our proposed framework classifies the points into seven challenging categories: *palm tree*, *cycas*, *brushwood*, *vehicle*, *light pole*, *grass*, and *road*.

The remainder of this paper is organized as follows. Section II introduces some related work. Section III presents the implementation details of the novel framework. Section IV reports and discusses the comparative experimental results on colorized mobile LiDAR point clouds acquired by RIEGL VMX-450 system. Finally, Section V gives the concluding remarks.

II. Related Work

In the last few years, research on semantic labeling of road scene mainly focused on 2D images [12]–[18]. Kang *et al.* [12] exploited multiband images to address the semantic segmentation tasks of road scenes by introducing a hierarchical bag-of-textons method to consider a pixel's larger visual context. In [17], a semantic labeling method was proposed for image sequences captured by a camera mounted on a car driving along streets. Based on these image sequences, spatial information of road scenes was explored by 3D scene reconstruction with the Structure from Motion [21] algorithm. The work [18] presented a novel framework for (1) enforcing temporal consistency between consecutive video frames and (2) considering locally reconstructed 3D geometry information when conducting semantic labeling tasks of road scenes. In [13], [14], nonparametric approaches were introduced to address the multiclass pixel labeling problem. Transferring annotations from labeled to unlabeled image sets, these nonparametric approaches accomplish promising labeling results on 2D road scene images. In spite of the success of on-image semantic labeling, the problems caused by object occlusions and image distortions still exist.

Semantic labeling of 3D point cloud road scenes has become an important problem because of its extensive use. Probabilistic graphical models such as Conditional Random Field (CRF) were widely exploited to solve this problem [22]–[27]. The CRF with a pairwise model was applied to ensure category label consistency between neighboring points [22], [26]. In [24], the Potts model [28], a high-order graphical model, was used to keep labels homogeneous in a clique. To allow a portion of inhomogeneous labels in a clique [23], Munoz introduced robust Potts model [29] into Max-Margin Markov Network (M3N). In [27], a set of non-associative context patterns were devised to describe geometric relationships between different category labels within the cliques. It is difficult for these CRF based methods to train a good model when incomplete objects exist in complex road scenarios. Some other methods did not use graphical models [30], [31]. In [30], a shape-based method was to (1) segment objects out of the point cloud scenes and (2) use global features to recognize objects in point cloud road scenes. Yang proposed a segmentation method to classify point clouds according to their geometric features and extract objects by merging the similar points and segments based on a set of empirical rules [31]. The performance of the two methods is influenced by segmentation results. Moreover, without local information, only global object information is not suitable for classifying these incomplete objects caused by occlusion and overlapping in point cloud road scenes.

III. Proposed Framework

As illustrated in Fig. 2, the proposed framework contains two stages: training and labeling stages. The training stage is carried out as follows: (1) extracting 3D patches from colorized mobile LiDAR point clouds and representing each 3D patch with rotation-invariant features, spatial coordinates, a category label, and a scene identifier (Section III-A); and (2) constructing a 3D-PMG by establishing correspondences between labeled 3D patches from different scenes (Section III-B). At the labeling stage, the trained 3D-PMG is first used to transfer category labels from labeled to unlabeled 3D patches (Section III-C). Then, in order to rectify the incorrect labeling of 3D patches, which is mainly caused by local 3D patch similarity in different categories, a MRF model is integrated into the proposed framework to consider contextual information (Section III-D).

A. 3D Patch Extraction and Description

There is a high point density of colorized mobile LiDAR point clouds acquired by mobile LiDAR systems, which results in computational burdens for labeling such huge numbers of points into specific categories. To alleviate such a burden, instead of labeling individual points, in this paper, 3D patches are extracted from the point clouds and treated as operating units for multi-category labeling tasks. Moreover, by using 3D

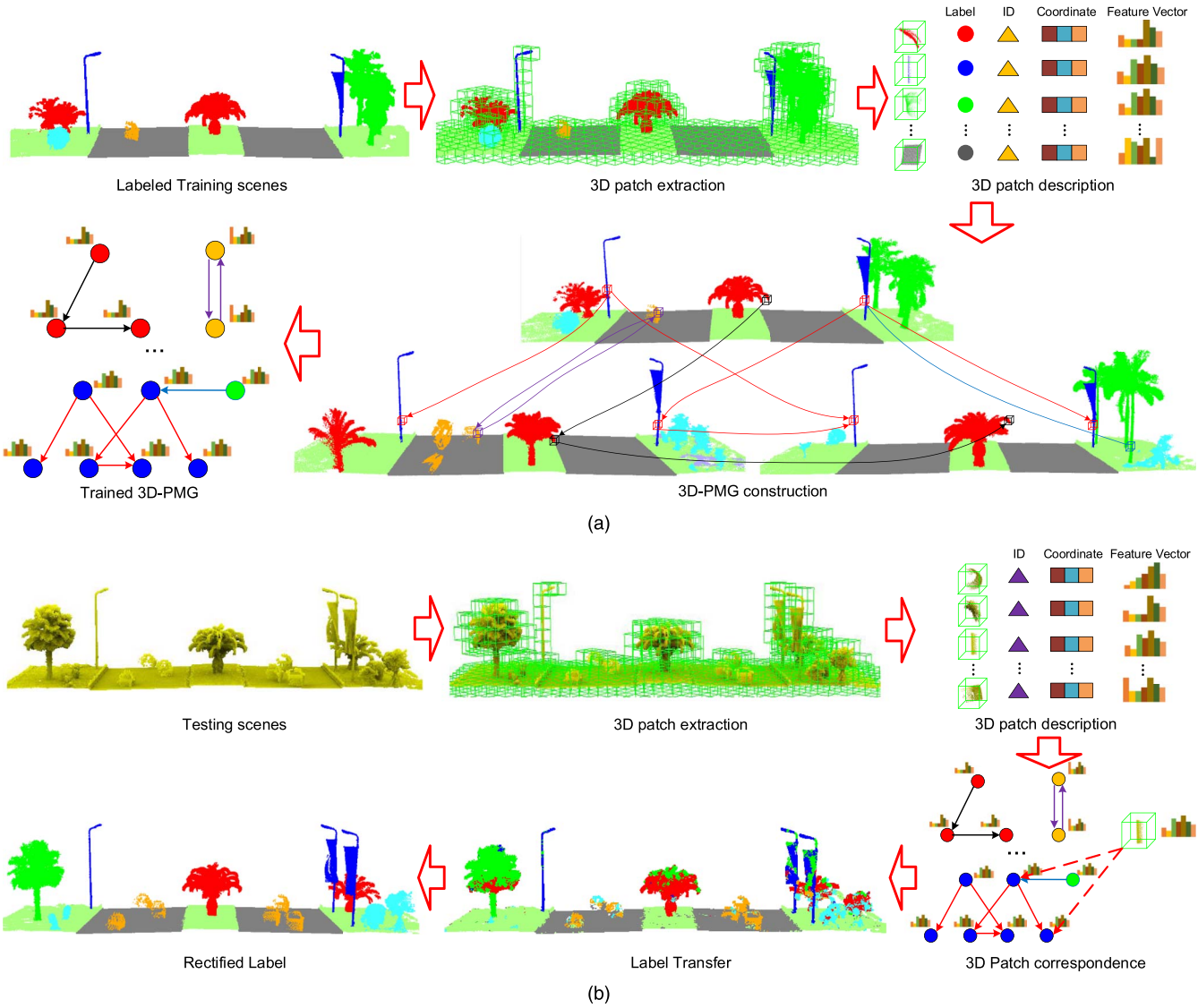


Fig. 2. Proposed semantic labeling framework: (a) training stage, (b) labeling stage. In 3D patch description, the cycle represents the category label and color indicates different categories. The triangle denotes the identifier of scene and different colors show different scenes. The rectangle denotes the coordinates of the centroid of the points containing in a 3D patch.

local patches to describe local features of objects, the problem for labeling incomplete objects might be alleviated.

For extracting 3D local patches, an Octree-based method is adopted. Specifically, after applying the Octree algorithm to partition point clouds spatially, each non-empty leaf node in the Octree structure [32] is treated as a 3D patch (see Fig. 2). To meet the requirements of the 3D-PMG (see details in Section III-B), each 3D patch is described with the following four kinds of information: category label, scene identifier, spatial coordinates, and feature description. Category label is selected as the majority label with labeled points in a 3D patch. At the labeling stage, the category labels of 3D patches are unknown in unlabeled scenes and can be inferred through label transfer. To distinguish 3D patches from different scenes, a unique identifier for each scene is assigned to 3D patches. To achieve rotation invariance, rotation-invariant feature descriptors are chosen to depict the geometric and texture features of a 3D patch. In our approach, Fast Point Feature Histograms (FPFH) [33] descriptors, the mean vector of the RGB channels, and the height information of the centroid of the points in a patch are exploited to describe shape, texture, and spatial information of a local patch, respectively.

B. 3D-PMG Construction

The 3D-PMG construction algorithm, an extension of the PatchMatchGraph algorithm [14] that is used for semantic segmentation in 2D images, is developed for transferring labels from labeled to unlabeled colorized mobile LiDAR point cloud scenes. The core of 3D-PMG construction algorithm is to find the approximate k-nearest neighboring 3D patches in the other scenes for each 3D patch extracted from each scene, while minimizing the cost between corresponding patches. To realize it, the 3D-PMG construction algorithm constructs a 3D-PMG via a set of move searches that exploit intrinsic and contextual properties of objects to effectively and efficiently

propagate 3D local patch matches among scenes. Combining with the local object information that is represented by 3D local patch, the intrinsic and contextual properties of objects assist in handling incompleteness of objects when transferring labels. To effectively organize these 3D patches, the Octree and KDTree algorithms are applied. To alleviate the challenge of category imbalance, the prior information such as category label is introduced into the proposed algorithm to guide move searches. Before introducing the implementation details of the 3D-PMG construction algorithm, we first define 3D-PMG and the objective function for constructing a 3D-PMG.

1) Definition and Objective Function: The 3D-PMG is formally defined over a set of 3D patches extracted from different colorized LiDAR point cloud scenes. The definition of 3D-PMG is as follows:

Definition 1: Let $S = \{S_1, \ldots, S_{n_s}\}$ be a set of n_s colorized LiDAR point cloud scenes and L be the size of each 3D patch. A 3D-PMG defined over S is a directed graph $\mathcal{G}^L(S) = \langle \mathcal{V}, \mathcal{E} \rangle$, where each node $u \in \mathcal{V}$ represents a distinct 3D patch extracted from one of the scenes in S and each edge $(u, v) \in \mathcal{E}$ denotes a match from node u to node v.

To measure the quality of the match between node u and node v, a cost function $c : \mathcal{E} \to \mathcal{R}$ is introduced. From the perspective of minimizing the cost of 3D patch matches, the **objective function** for constructing a 3D-PMG is as follows:

$$\begin{aligned} \min \quad & \sum_{(u,v)\in\mathcal{E}} c(u, v) \\ \text{s.t.} \quad & \forall u \in \mathcal{V} : \mathbf{deg}(u) = K \\ & \forall (u, v) \in \mathcal{E} : S_{(u)} \neq S_{(v)} \\ & \forall (u, v),\ (u, w) \in \mathcal{E} \\ \text{s.t.} \quad & v \neq w : S_{(v)} \neq S_{(w)} \end{aligned} \tag{1}$$

where $\mathbf{deg}(u)$ is the outdegree of node u; $S_{(u)}$ represents the scene to which node u belongs; $c(u, v)$ is the matching cost between u and v. The cost function $c(u, v)$ is defined as follows:

$$c(u, v) = \alpha_c \cdot c_c(u, v) + \alpha_f \cdot c_f(u, v) + \alpha_h \cdot c_h(u, v) \tag{2}$$

$$c_c(u, v) = \sum_{k=1}^{3} \frac{|C_u(k) - C_v(k)|}{255} \tag{3}$$

$$c_f(u, v) = \sum_{k=1}^{16} \frac{[F_u(k) - F_v(k)]^2}{F_u(k) + F_v(k)} \tag{4}$$

$$c_h(u, v) = \frac{|H_u - H_v|}{H_{\max}} \tag{5}$$

where α_c, α_f, and α_h are the cost weights for $c_c(u, v)$, $c_f(u, v)$, and $c_h(u, v)$, respectively. $\mathbf{F}_u$ denotes a 16-D FPFH descriptor for a 3D patch u; $\mathbf{C}_u$ represents a RGB color vector of a 3D patch u; H_u represents height information of a 3D patch u; $H_{\max}$ is a constant to normalize the height information. Eq. (4) computes the shape similarity cost between 3D patches u and v using the χ^2 distance [34].

The requirement on matches being from different scenes results in a diversity of matches. Such diversity in matches is important to ensure our proposed algorithm to escape local optima quickly. Therefore, the objective function implies a set of constraints to ensure a diversity of matches. These constraints include: (1) 3D patches extracted from the same scene will not be matched for avoiding a 3D patch matching itself; (2) A 3D patch is constrained to match at most one 3D patch from the same scene; (3) A 3D patch should ascertain a total of K matches from all point cloud scenes.

The huge number of 3D patches results in high computational complexity and large memory requirements to find the optimal solution of Eq. (1). Instead of an optimal solution, an approximate optimal solution is obtained by applying the 3D-PMG construction algorithm.

2) 3D-PMG Algorithm: The 3D-PMG construction algorithm needs to search 3D patch matches under the constraints in Eq. (1). For the purpose of searching good candidate matches, the algorithm defines a set of match rules called search moves which contain: initialization, propagation, local search, inverse and forward enrichments, and exhaustive search. The details are explained in the following:

Initialization: There are two methods for initialization. (1) Graph $\mathcal{G}$ is initialized by each node $u \in \mathcal{V}$ with K random matches under the constraints in Eq. (1). Specifically, $u \in \mathcal{V}$ is firstly initialized with all valid 3D patches extracted from point cloud scenes and $\mathcal{E}$ is set to be an empty set. Then, for each node $u \in \mathcal{V}$, K nodes $v_1, \ldots, v_K \in \mathcal{V}$ are randomly selected from K different scenes, and K edges (u, v_i), $i = 1, \ldots, K$ are added to $\mathcal{E}$. (2) To make the algorithm converge quickly, height information is introduced as an indicator to conduct the initialization. Concretely, for each node $u \in \mathcal{V}$, firstly, K different scenes are randomly selected. Then, for each scene, a node v is randomly selected among the 3D patches with approximately identical height to u. Finally, the edge (u, v) is added to $\mathcal{E}$. The proposed approach combines these two methods and selects the current best K matches satisfying the constraints in Eq. (1). Initialization is performed only once.

Propagation: In 2D images, it is observed that if a patch u is a good match with patch v, then this implies that spatially adjacent patches to u are also good matches with spatially adjacent patches to v [35]. This observation also works well in colorized mobile LiDAR point cloud scenes. However, neighboring relationship of 3D patches in point cloud scenes is difficult to define efficiently compared with images which define 4-connected pixels as the neighbors. Thus, the proposed framework exploits the KDTree algorithm to organize these 3D patches. Neighbors of v are defined as those 3D patches whose spatial distances to v are less than a searching radius R_{pro}. After that, as illustrated in Fig. 3(a), for each neighbor v' of v and each neighbor u' of u, edges (u', v') are considered as good candidate matches. This move propagates good matches across the space of 3D patches.

Local search: The local search move shown in Fig. 3(b) aims to search a better match locally. Considering an existing edge $(u, v) \in \mathcal{E}$, for all 3D patches v' whose spatial distance to v is less than the radius R_{loc}, edges (u, v') are considered as good candidate matches. Local search costs less time than propagation.

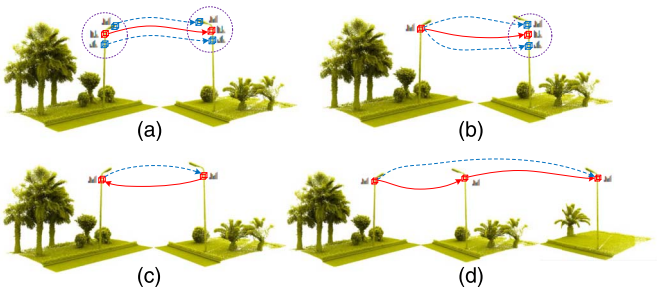


Fig. 3. Search moves: (a) propagation, (b) local search, (c) inverse enrichment, (d) forward enrichment. Each cube box represents a 3D patch. The solid line represents the edge already existing in the graph. The dashed line represents the candidate edge.

Inverse enrichment: Fig. 3(c) depicts the procedure of inverse enrichment. The inverse enrichment is motivated by the observation that if node v is a good match to node u, then inversely, node u may be a good match to node v. Thus, for an edge (u, v) in $\mathcal{E}$, edge (v, u) is also regarded as a good candidate match. To prevent cycles (and enforce constraints in Eq. (1)), we skip edges that match back to the same scene.
Forward enrichment: As illustrated in Fig. 3(d), for forward enrichment, if there are edges (u, v) and $(v, w) \in \mathcal{E}$, then edge (u, w) is a good candidate match. This move strategy makes the good matches likely to spread over the entire graph.
Exhaustive search: Exhaustive search, which assists in escaping local optima, is a brute-force search method. For a node $u \in \mathcal{V}$ in a scene, all edges (v, u), whose node v come from other scenes, are considered as good candidate matches. This move requires much more time than the aforementioned search moves. In the proposed framework, a few nodes are selected to conduct an exhaustive search during the 3D-PMG construction procedure. Many alternative strategies can be used to select nodes for exhaustive search. The proposed framework introduces the following two strategies: (1) A node is selected with a probability proportional to the cost of its best current match. This helps the 3D-PMG improve poor matches. (2) The prior information, category label, is chosen as an indicator to select nodes. Because, in most cases, the 3D patches extracted from some objects, which account for minority categories in the scenes, are inclined to fall into local optima, thereby resulting in a failure to find good matches. Thus, to solve such a problem caused by category imbalance, we randomly select the 3D patches from these minority categories to perform an exhaustive search.

Algorithm 1 Updating a 3D-PatchMatchGraph with a candidate match

Input: old graph $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$ and new edge $e = (u, v)$
Output: $\mathcal{G}^*$

1: Let $e' = \arg\max_{(u,v') \in \mathcal{E}} \begin{cases} \infty & \text{if } S_{(v')} = S_{(v)} \\ c(u, v') & \text{otherwise} \end{cases}$
2: **if** $c(u, v) < c(u, v')$ **then**
3: $\quad \mathcal{E} \leftarrow \mathcal{E} \bigcup \{e\} \setminus \{e'\}$
4: **end if**
5: **return** $\mathcal{G}^* = \langle \mathcal{V}, \mathcal{E} \rangle$

The 3D-PMG construction algorithm is carried out as follows. First, a 3D-PMG is initialized with matches satisfying the constraints in Eq. (1). Then, we start iterating. In each iteration, all the nodes whose matches have changed in the previous iteration are selected. The search moves including propagation, local search, forward enrichment, and inverse enrichment, are conducted for the changed edges associated with these nodes. In addition, a few nodes are selected to perform an exhaustive search in each iteration. Each of these search moves tries to find candidate matches and progressively updates the matches in the 3D-PMG. The update is accomplished by using Algorithm 1. Search moves are terminated when an iteration reaches a maximum or the value of the objective function (see Eq. (1)) no longer changes. The procedure of applying 3D-PMG construction algorithm at training stage is described as follows: as shown in Fig. 2(a), for a set of labeled colorized LiDAR point cloud scenes, firstly, labeled 3D patches are extracted and characterized. Then, by using these labeled 3D patches, a 3D-PMG is trained or built through the proposed 3D-PMG construction algorithm.

C. Label Transfer With 3D-PMG

Transferring category labels from labeled to unlabeled scenes is carried out at the labeling stage. As illustrated in Fig. 2(b), for an unlabeled scene, 3D patches extracted from unlabeled scenes are first described by three kinds of information: scene identifier, spatial coordinates, and feature description. Then these directional correspondences from unlabeled 3D patches to labeled nodes in the trained 3D-PMG are established by the 3D-PMG construction algorithm. Based on these 3D patches correspondences, label transfer is applied to infer the initial labels of the unlabeled patches by Eq. (6) and Eq. (7). The search moves applied in label transfer differ from those at the training stage in the following two aspects: on the one hand, during the establishment of the 3D patches correspondences, we consider only the 3D patches from unlabeled scenes; on the other hand, edges are unidirectional from the labeled 3D patches to the trained 3D-PMG, thereby, resulting in the removal of inverse enrichment in label transfer. The probability of an unlabeled node u taking a label l is estimated as follows:

$$P(y_u = l|\mathcal{G}) \propto \sum_{(u,v)\in\mathcal{E}} w_u^l(u, v)[L_v = l] \tag{6}$$

where $[\cdot]$ is an indicator function. The value of $[\cdot]$ is 1 if its argument is true, otherwise 0. L_v indicates the category label of node v. $w_u^l(u, v)$ is a weight function which adjusts the contribution of each match. There are many alternative methods to define such a weight function. Here, the contributions of matches are weighted by a means of their costs. We follow the work [14] to define the weight function as $1/r^2$, where r is the rank of the match in the sorted list of matches for node u.

The category label L_u of an unlabeled node u is inferred by following:

$$L_u = \arg\max_l P(y_u = l|\mathcal{G}). \tag{7}$$

To determine the category label of an individual 3D point p in a colorized LiDAR point cloud scene, we refer to p as a center point and R_{area} as a radius for searching the spatial neighboring nodes V_N of p by the KDTree algorithm. The category label l of point p is determined as follows:

$$l_p = \arg\max_{l} \sum_{v \in V_{N_p}} [L_v = l]. \tag{8}$$

D. Label Refinement With MRF

As shown by the point cloud labeling results in label transfer in Fig. 2(b), some incorrect labels are caused by local patch similarities in different categories. We use the MRF model to refine the labeling results with consideration of contextual information among 3D local patches. The Graph Cuts [36] algorithm is adopted to minimize the energy of the following pairwise MRF:

$$E(y_1, \ldots, y_{|\mathcal{V}|}) = \sum_{u \in \mathcal{V}} D_u(y_u) + \lambda \sum_{(u,v) \in \mathcal{E}_{\text{mrf}}} B_{u,v}(y_u, y_v) \tag{9}$$

where y_u is the label of a 3D patch u; $\mathcal{E}_{\text{mrf}}$ constrains all edges in the MRF; D is a unary energy term and B is a pairwise energy term. λ is a parameter for controlling the weight of the pairwise energy term.

The initial label probabilities of 3D patches $P(y_u|\mathcal{G})$ computed in label transfer are used to formulate the unary energy term D_u

$$D_u(y_u) = 1 - \log\ P(y_u|\mathcal{G}). \tag{10}$$

To group 3D patches into specific categories with appearance consistency and spatial adjacency, color information is used to design the pairwise energy term $B_{u,v}$. However, because of inter-category color similarities in our dataset, it is difficult to distinguish between *grass* and vegetation, *vehicle* and *ground* with only color information. In label transfer, *ground* is inclined to be incorrectly labeled as *grass* in most conditions and vice versa. Therefore, to solve the above issue, we incorporate the patch's label obtained in label transfer (initial label) into the design of the pairwise energy term $B_{u,v}$

$$B_{u,v}(y_u, y_v) = \begin{cases} 0, & y_u = y_v \\ \gamma, & y_u \neq y_v \wedge y'_u = l_{\text{grass}} \wedge y'_v \neq l_{\text{ground}} \\ \gamma, & y_u \neq y_v \wedge y'_u = l_{\text{ground}} \wedge y'_v \neq l_{\text{grass}} \\ \exp\left(-\beta \left\|C(u) - C(v)\right\|^2\right), & \text{otherwise} \end{cases} \tag{11}$$

where $C(u)$ represents the RGB color vector of node u. l_{grass} and l_{ground} are category labels of *grass* and *ground*, respectively. y'_u is the node u's initial label inferred in label transfer. γ is a penalty under the following two conditions: 1) the initial label of a 3D patch labeled as grass is altered into another category other than the ground, 2) the initial label of a 3D patch labeled as *ground* is altered into another category other than the grass. In our method, we simply set γ at zero. The pairwise MRF only has two parameters λ and β, which are readily set by cross-validations.

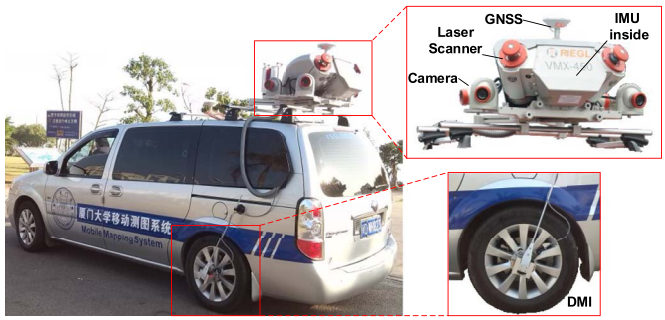


Fig. 4. Illustration of REIGL VMX-450 mobile LiDAR system and its configurations.

IV. Experiment and Discussion

A. RIEGL VMX-450 System and Colorized Mobile LiDAR Point Cloud Dataset

In this paper, the colorized mobile LiDAR point clouds were acquired along the Ring Road in Xiamen, China, by a REIGL VMX-450 mobile LiDAR system. The REIGL VMX-450 system [37] smoothly integrates two RIEGL VQ-450 laser scanners, a global navigation satellite system (GNSS) antenna, an inertial measurement unit (IMU), a distance measurement indicator (DMI), and four high-resolution digital cameras (see Fig. 4). This integrated set was mounted on the roof of a minivan with an average speed of 40–50 km/h. After data acquisition, RiProcess, a post-process software released by REIGL corporation, is used to calibrate the images with point clouds for the generation of colorized mobile LiDAR point clouds.

A colorized mobile LiDAR point cloud dataset with a point density of about 7,000 points/m^2 and covering a road section of about 1,500 meters was selected to evaluate the proposed framework. To evaluate the performance of our proposed framework, we built a ground truth for the selected dataset by manually and thoroughly classifying all points into the following eight categories: *road*, *grass*, *palm tree*, *cycas*, *brushwood*, *light pole*, *vehicle*, and *others*. Table I details the number of points in each category. In order to conveniently handle the huge number of points in the dataset, we partitioned it into 142 parts along the trajectory which was recorded by the onboard navigation system. Each part, with a length of about 10 meters and a size of about 40 MB, was treated as an individual scene. All the scenes were split into a training dataset with 20 scenes and a test dataset with 122 scenes. Challenges, such as objects overlapping, occlusions, incompleteness, and category imbalance, commonly existed in the dataset. Specifically, Table I presents the number of objects in each category and the number of overlapping objects in our dataset. The statistical data show the challenges of objects overlapping and category imbalance. Moreover, as illustrated in Fig. 5(f), the *light pole* and *palm trees* are overlapped, and the palm trees cluster. In Fig. 5(i), the vehicle is incomplete because of self-occlusions. These challenges make the road scene labeling of mobile LiDAR point clouds difficult.

B. Semantic Labeling of Road Scenes

A 3D-PMG is trained by using the 20 training scenes in our dataset. Firstly, 3D patches are extracted from each training

TABLE I
DESCRIPTION AND GROUND TRUTH OF THE DATASETS

	road	grass	palm tree	cycas	brushwood	light pole	vehicle	other
points	90,561,689	28,429,942	8,131,790	4,611,530	3,175,866	658,189	479,799	205,850
objects	-	-	278	131	220	90	99	-
overlapping objects	-	-	240	65	50	7	0	-

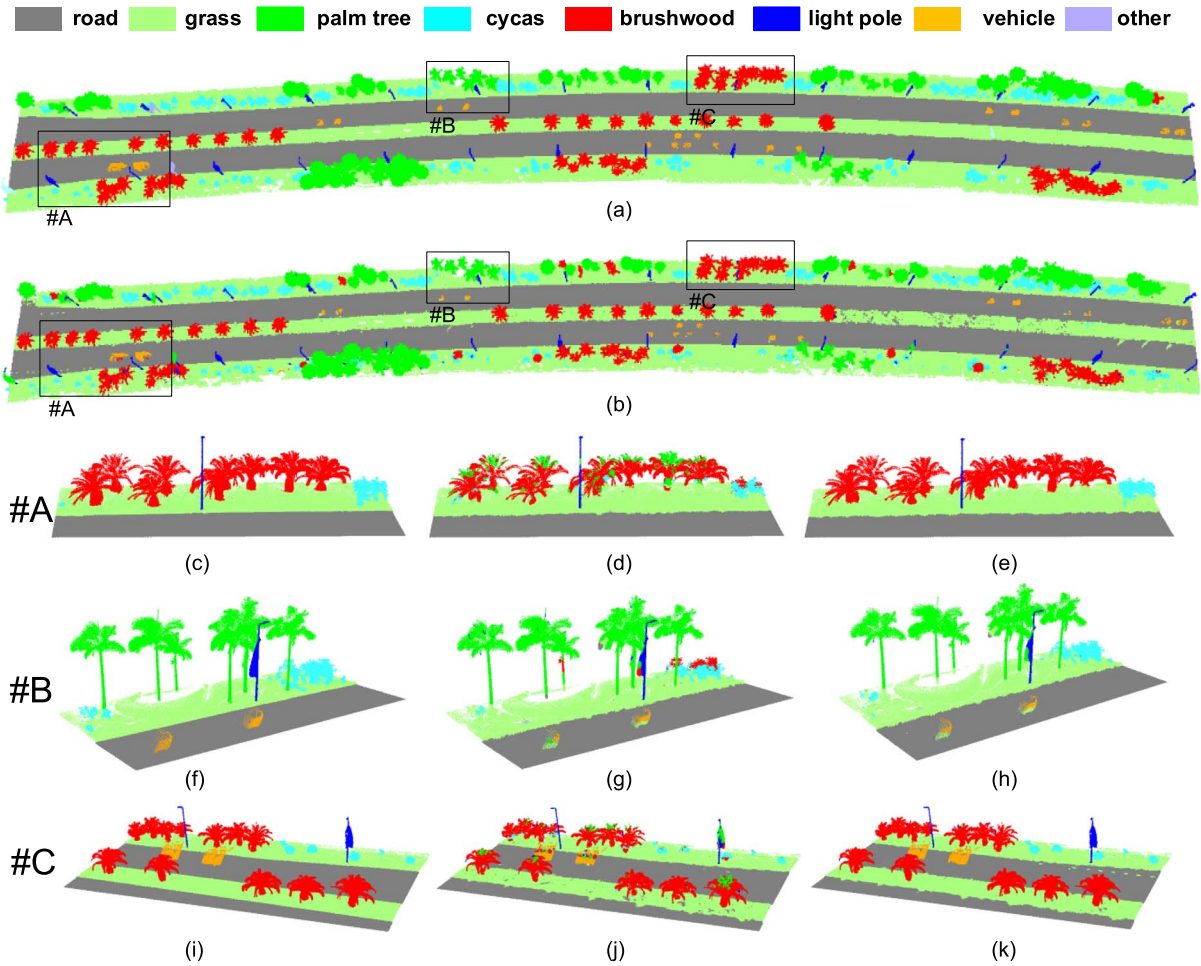


Fig. 5. Semantic labeling results on a part of our dataset: (a) ground truth; (b) semantic labeling results; (c), (f), and (i) close-up views of the ground truths in areas #A, #B, and #C; (d), (g), and (j) close-up views of the label transfer results in areas #A, #B, and #C; (e), (h), and (k) close-up views of the final semantic labeling results in areas #A, #B, and #C.

scene with the same size, L, and each patch is as the illustration in Section III-A. Then the 3D-PMG construction algorithm is exploited to establish correspondence relationships among these 3D patches and finally construct a trained 3D-PMG. In the experiments, an exhaustive search, with the two following strategies, is implemented by: 1) randomly selecting 0.1% 3D patches in minority categories, including *vehicle*, *cycas*, and *light pole*; 2) selecting ten 3D patches with a probability proportional to the cost of the best current match in each iteration. Search moves mentioned in Section III-B are all included in the training stage. During construction of 3D-PMG, we set iteration times to be 50. Several 3D-PMG construction related parameters and their corresponding values are listed in Table II. The two parameters λ and β in the MRF model are obtained by applying a grid search method.

At the labeling stage, the 122 scenes in the test dataset were used for evaluating the performance of the proposed framework. After establishing directional correspondences from unlabeled 3D patches to the labeled nodes in the trained 3D-PMG, the initial label of each unlabeled 3D patch was inferred from Eq. (6) and Eq. (7). We set R_{area} as 0.3 m to determine neighboring 3D patches for individual points and generated label transfer results for individual points with Eq. (8). By applying MRF, the refined labeling results are obtained.

TABLE II
PARAMETERS IN 3D-PMG CONSTRUCTION

$L(m)$	$R_{loc}(m)$	$R_{pro}(m)$	α_c	α_h	α_f	K	$H_{max}(m)$
0.2	1.0	0.5	0.3	0.3	0.4	7	20

The label transfer results of the road scenes are shown in Fig. 5(d), (g), and (j). Although there are overlapping objects and incomplete objects in the road scenes, the local patch-based graph structure (3D-PMG) effectively exploits local

TABLE III
EXPERIMENTAL RESULTS ON DIFFERENT APPROACHES

		road	grass	palm tree	cycas	brushwood	light pole	vehicle	argv
Precision	SVM-based	.95	.839	.677	.745	.61	.726	.489	.719
	RF-based	.965	.71	.708	.83	**.624**	**.854**	.678	.767
	Shape-based [30]	-	-	.907	.648	.136	.731	.56	.596
	M3N [23]	.971	.933	.891	.80	.538	.526	.65	.759
	3D-PMG (ours)	**.975**	.95	.856	.802	.481	.682	.478	.746
	3D-PMG + MRF(ours)	.971	**.955**	**.96**	**.926**	.564	.742	**.762**	**.84**
Recall	SVM-based	.987	.846	.939	.595	.315	.289	.387	.623
	RF-based	.986	.862	.931	.598	.229	.362	.433	.629
	Shape-based [30]	-	-	.773	.532	.455	**.958**	.598	.663
	M3N [23]	.988	.876	.867	.718	.624	.862	.852	.827
	3D-PMG (ours)	.986	.881	.913	.678	.761	.697	.772	.813
	3D-PMG + MRF(ours)	**.989**	**.883**	**.962**	**.793**	**.81**	.881	**.956**	**.896**
F1-measure	SVM-based	.968	.842	.787	.662	.416	.414	.432	.646
	RF-based	.976	.778	.804	.696	.335	.509	.528	.66
	Shape-based [30]	-	-	.835	.585	.209	**.83**	.578	.607
	M3N [23]	**.98**	.904	.879	.757	.578	.653	.737	.784
	3D-PMG (ours)	**.98**	.914	.884	.735	.59	.689	.59	.769
	3D-PMG + MRF(ours)	**.98**	**.918**	**.961**	**.855**	**.665**	.805	**.847**	**.862**

information, intrinsic and contextual properties of objects to transfer category labels. In addition, for a small portion of incorrect labels caused by the local similarities in different categories (e.g. leaves of *cycas* and *palm tree*), MRF considers contextual information to refine the labeling results as shown in Fig. 5(e), (h), and (k)). Fig. 5(b) presents a part of the semantic labeling results on the test road scenes, which shows that our proposed framework performs promising semantic labeling results on road scenes.

To quantitatively assess the accuracy and correctness of the semantic labeling results on the test dataset, we selected the following three measures including Precision, Recall, and F1-measure [27]. Precision describes the percentage of true positives in the ground truth; recall depicts the percentage of true positives in the semantic labeling results; F1-measure is an overall measure. The three measures are calculated on points and defined as follows:

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{12}$$

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{13}$$

$$F1 - \text{measure} = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \tag{14}$$

where TP, FN, and FP represent the number of true positives, false negatives, and false positives, respectively. The quantitative evaluation results using these three measures are presented in Table III. The average label transfer results (3D-PMG) achieved in precision, recall, and F1-measure are 0.746, 0.813, and 0.769, respectively. The proposed framework (3D-PMG+MRF) achieves an average precision, recall, and F1-measure of 0.84, 0.896, and 0.862, respectively, in labeling road scenes. There is an improvement of performance by incorporating MRF model into our proposed framework, because the contextual information among 3D patches is considered by using MRF model.

C. Comparative Studies

To show the superiority of the proposed framework (3D-PMG+MRF) in labeling road scenes, the performance of the label transfer part in our proposed framework is first evaluated. For comparing it with two classic classifiers: Random Forest (RF) [38] and Support Vector Machine (SVM) [39], we treated the label transfer part as a classifier in multi-category classification of colorized mobile LiDAR point clouds. For the SVM-based method, we used the LIBSVM [40] and selected radial basis function as the kernel function. For the RF-based method, we set the number of trees at ten and the depth of each tree at fifteen. For RF and SVM classifiers, the size and the feature descriptors of 3D local patches were the same as the 3D-PMG-based method. To balance the training samples in each category, we randomly sampled 10,000 patches for each category from the training dataset to train the SVM and the RF classifiers. After inferring the category labels of 3D patches in the test dataset, the category label of each single point was determined by Eq. (8). As shown by the quantitative evaluations in Table III, compared with the SVM-based and RF-based methods, label transfer by applying 3D-PMG algorithm obtains the highest average F1-measure of 0.769 and precision of 0.813. Although the SVM-based and the RF-based methods achieved better results in precision in classifying points of some certain categories, including brushwood, light pole, and vehicle, our method (3D-PMG) attained the highest recall and F1-measure in each category. This is because, compared with the SVM-based and the RF-based methods that exploit only local object information, label transfer in our proposed framework is performed by, not only local object information, but also by the objects intrinsic and contextual properties.

To further evaluate the performance of the proposed framework (3D-PMG+MRF) in labeling road scenes, the M3N approach [23] and the shape-based approach [30] were selected for comparison. The M3N approach adopted the robust Potts model [29] where the cliques were defined to be the resulting clusters from two k-means segmentations [41] with parameter

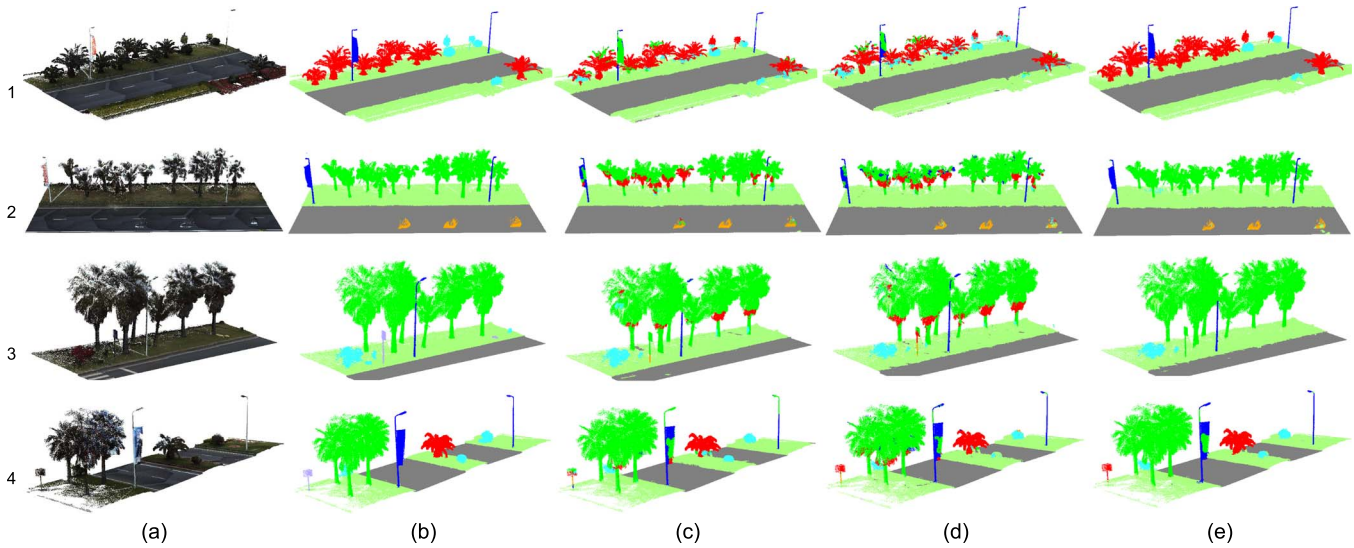


Fig. 6. Semantic labeling results in comparison: (a) colorized mobile LiDAR point clouds; (b) ground truths; (c) results of the M3N approach; (d) results of label transfer; (e) results of 3D-PatchMatchGraph with pairwise MRF.

configurations of $k_1 = 0.03$ and $k_2 = 0.01$, respectively. Considering the point density and the huge number of points, we adopted the 3D patch extraction strategy aforementioned in Section III-A to downsample the points. The shape-based approach included the following four steps: locating, segmenting, characterizing, and classifying clusters of 3D points. As the quantitative results in Table III shown, our proposed framework yields an average F1-measure of 0.862, which outperforms both two methods: the shape-based methods whose average F1-measure is only 0.607, and the M3N method whose average F1-measure is 0.784. The poor performance of the shape-based approach demonstrates that it is not suitable for these scenarios with severely overlapping and incomplete objects. The M3N method outperforms the other methods except for our proposed framework, and as seen in Fig. 6, the labeling results generated by the M3N approach are not as smooth as it in our proposed framework. This is because it is difficult for the M3N method to train a probabilistic graphical model to fit complex scenarios with unbalanced categories and incomplete objects.

The proposed framework and comparative studies were coded with C++ and executed on a personal computer with a single Intel core of 3.30 GHz and a RAM of 16 GB. The training times for SVM-based, RF-based, Shape-based and M3N methods were 1.5, 0.5, 0.8, and 2.4 hours, respectively. The labeling times of the whole test dataset for SVM-based, RF-based, Shape-based and M3N methods were 4.4, 3.8, 4.2, and 5.8 hours, respectively. For our proposed framework, the training time was about 5.5 hours and the labeling time for the whole test dataset was about 8.6 hours. The average time for labeling an individual scene was about 4 minutes. Although our proposed framework took more time than those of the other methods, it achieved a superior labeling performance than the other methods. Moreover, in the 3D-PMG construction algorithm, search moves such as propagation, inverse enrichment, and local search moves can be implemented in parallel. The computational cost of these moves can be reduced to about $1/k$ times of the non-parallel execution time. Here, k is the number of cores.

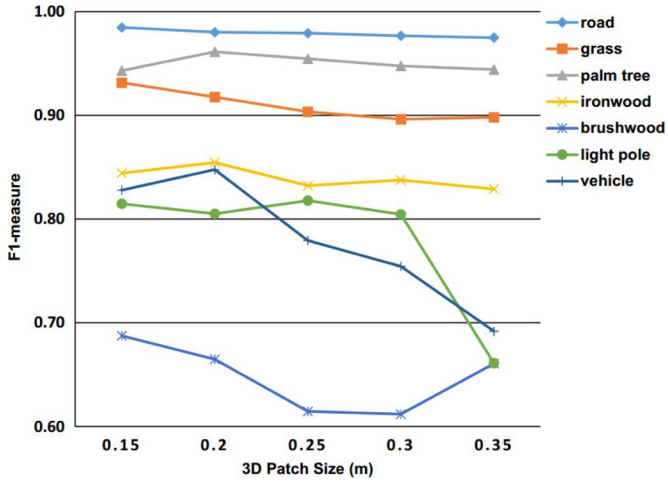


Fig. 7. Impact of 3D patch size on semantic labeling results.

D. Sensitivity of Our Proposed Framework

In this section, we analyze the impact of the following parameters on the performance of the proposed approach: size of 3D patch (L), and local search radius (R_{loc}).

To analyze the influence of 3D patch size on the proposed approach, the following configurations were tested: 0.15 m, 0.2 m, 0.25 m, 0.3 m, and 0.35 m. The test results are shown in Fig. 7. As seen in Fig. 7, the F1-measures for *road*, *grass*, *palm tree*, and *cycas*, change slightly as the 3D patch size increases. However, for *vehicles*, the F1-measure drops from 0.85 to 0.69 when the 3D patch size ranges from 0.2 m to 0.35 m. For *light poles*, the F1-measure decreases dramatically from $L = 0.3$ m to $L = 0.35$ m. In fact, the number of 3D patches will decrease with the increase of 3D patch size. As a result, for those categories with minor number of points, the possibility of matching correct 3D patches decreases as the 3D patch size increases. Moreover, a large local 3D patch is not suitable for describing simple structures like surfaces or lines because of high possibility to introduce noisy points which

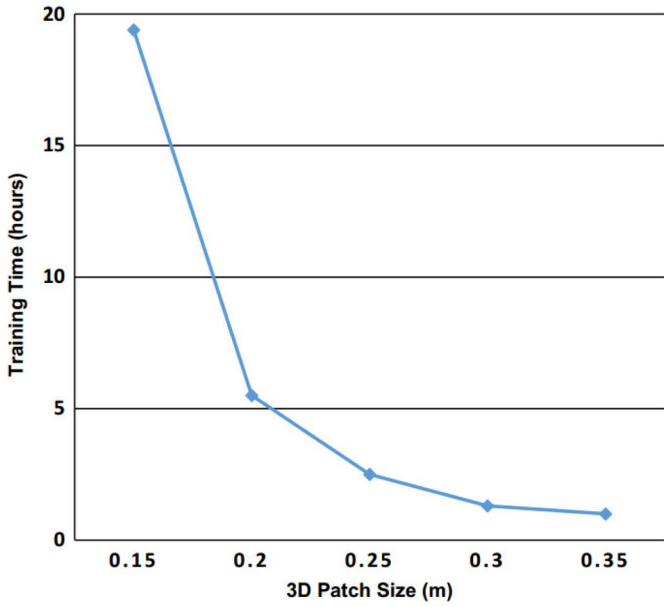


Fig. 8. Impact of 3D patch size on training time.

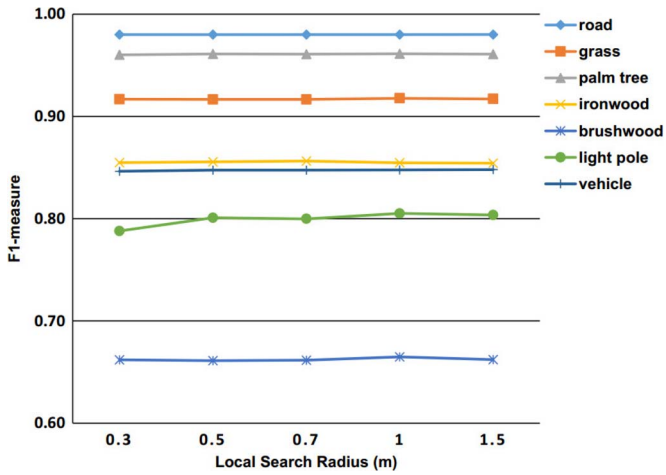


Fig. 9. Impact of local search radius on semantic labeling results.

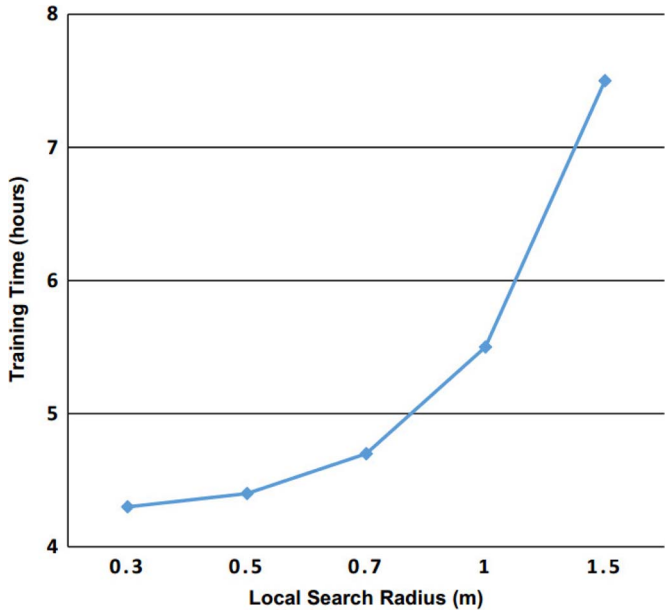


Fig. 10. Impact of local search radius on training time.

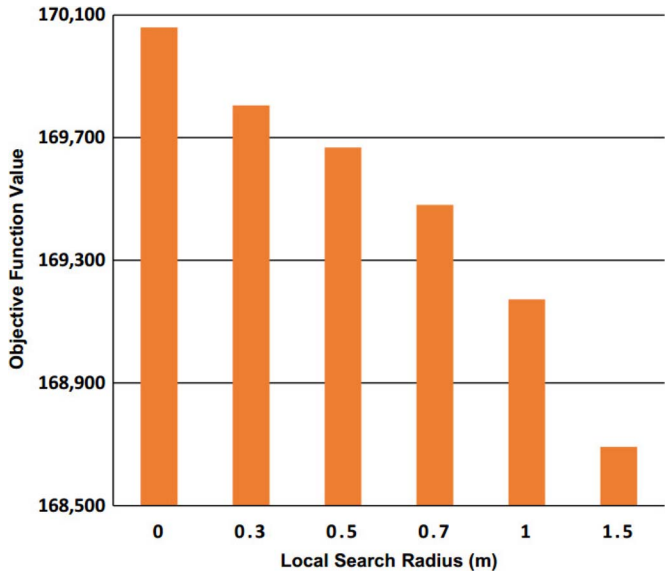


Fig. 11. Impact of local search radius on objective function value.

destroy these simple structures. As shown in Fig. 8, as the 3D patch size increases, the training time decreases. This is because the training time is influenced by the number of 3D patches. To make a trade-off between the performance and the efficiency, we set the 3D patches size at $L = 0.2$ m.

To analyze the impact of local search radius on the performance of our proposed approach, the following configurations were tested: 0.3 m, 0.5 m, 0.7 m, 1 m, and 1.5 m. As reflected in Fig. 9, the F1-measures change very slightly with an increase in the local search radius. This is because propagation achieves a similar effect to local search at the expense of a low convergence rate. The reason for conducting a local search in the proposed framework is that a larger local search radius ascertains good matches of the current node rapidly. In addition, as reflected in Fig. 11, the values of the objective function in Eq. (1) decreases as the local search radius increases at the training stage, which shows that the larger local search radius assists in finding better local matches. Moreover, the training time varies with an increase in local search radius (see Fig. 10). This is because the larger search radius requires more time to search for candidate matches. To make a trade-off between the performance and the efficiency of our proposed framework, we set the local search radius at 1.0 m (see Fig. 11).

V. Conclusion

In this paper, in order to effectively conduct semantic labeling tasks on road scenes, we have presented a patch-based framework that combines a 3D-PMG structure and a pairwise MRF model by using colorized mobile LiDAR point clouds. The 3D-PMG was used to transfer category labels from labeled to unlabeled 3D patches by exploiting both intrinsic and contextual properties rather than local features only. The pairwise MRF model was exploited to refine the label transfer results with contextual information. For evaluating the proposed

framework, a ground truth with challenges including incomp lete objects, overlapping objects, and inter-class local similarities, was built. Quantitative evaluations demonstrated that our proposed framework achieves an average precision, recall, and F1-measure of 0.84, 0.896, and 0.862, respectively. In addition, comparative studies also demonstrated the superior performance of our proposed algorithm over the M3N method and the shape-based method in semantically labeling complex road scenes. Although our proposed framework obtained superior performance, the labeling results still have space to be improved by introducing higher-order [42] or co-occurrence potentials [43] into the MRF model. We will try more complex potentials to improve the labeling performance in our future work.

In addition, because of expensive computational costs, the proposed framework is feasible and promising to some off-line road labeling tasks. With the success in road scene labeling based on colorized mobile LiDAR point clouds, the accurate position and category information of objects may assist in ITS-related applications to well interpret the road environments.

REFERENCES

[1] C. Guo, J. Meguro, Y. Kojima, and T. Naito, "A multimodal ADAS system for unmarked urban scenarios based on road context understanding," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 4, pp. 1690–1704, Aug. 2015.

[2] G. Singh and J. Kosecka, "Acquiring semantics induced topology in urban environments," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2012, pp. 3509–3514.

[3] A. Geiger, C. Wojek, and R. Urtasun, "Joint 3d estimation of objects and scene layout," in *Proc. Conf. Adv. Neural Inf. Process. Syst.*, 2011, pp. 1467–1475.

[4] C. Fernández, R. Izquierdo, D. Llorca, and M. Sotelo, "Road curb and lanes detection for autonomous driving on urban scenarios," in *Proc. Int. IEEE Annu. Conf. Intell. Transp. Syst.*, 2014, pp. 1964–1969.

[5] F. Oniga, S. Nedevschi, and M. M. Meinecke, "Curb detection based on a multi-frame persistence map for urban driving scenarios," in *Proc. Int. IEEE Annu. Conf. Intell. Transp. Syst.*, 2008, pp. 67–72.

[6] F. Oniga and S. Nedevschi, "Curb detection for driving assistance systems: A cubic spline-based approach," in *Proc. IEEE Intell. Veh. Symp.*, 2011, pp. 945–950.

[7] G. Maier, S. Pangerl, and A. Schindler, "Real-time detection and classification of arrow markings using curve-based prototype fitting," in *Proc. IEEE Intell. Veh. Symp.*, 2011, pp. 442–447.

[8] J. C. McCall and M. M. Trivedi, "Video-based lane estimation and tracking for driver assistance: Survey, system, and evaluation," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 1, pp. 20–37, Mar. 2006.

[9] Y. He, H. Wang, and B. Zhang, "Color-based road detection in urban traffic scenes," *IEEE Trans. Intell. Transp. Syst.*, vol. 5, no. 4, pp. 309–318, Dec. 2004.

[10] H. Guan, J. Li, Y. Yu, Z. Ji, and C. Wang, "Using mobile LiDAR data for rapidly updating road markings," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 5, pp. 2457–2466, Oct. 2015.

[11] R. Marc, G. Dominique, and P. Evangeline, "Generator of road marking textures and associated ground truth applied to the evaluation of road marking detection," in *Proc. Int. IEEE Annu. Conf. Intell. Transp. Syst.*, 2012, pp. 933–938.

[12] Y. Kang, K. Yamaguchi, T. Naito, and Y. Ninomiya, "Multiband image segmentation and object recognition for understanding road scenes," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 4, pp. 1423–1433, Dec. 2011.

[13] C. Liu, J. Yuen, and A. Torralba, "Nonparametric scene parsing via label transfer," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 12, pp. 2368–2382, Dec. 2011.

[14] S. Gould and Y. Zhang, "Patchmatchgraph: Building a graph of dense patch correspondences for label transfer," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 439–452.

[15] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, "Learning hierarchical features for scene labeling," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1915–1929, Aug. 2013.

[16] G. J. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla, "Segmentation and recognition using structure from motion point clouds," in *Proc. Eur. Conf. Comput. Vis.*, 2008, pp. 44–57.

[17] J. Xiao and L. Quan, "Multiple view semantic segmentation for street view images," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2009, pp. 686–693.

[18] G. Floros and B. Leibe, "Joint 2D–3D temporally consistent semantic segmentation of street scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 2823–2830.

[19] K. Williams, M. J. Olsen, G. V. Roe, and C. Glennie, "Synthesis of transportation applications of mobile LiDAR," *Remote Sens.*, vol. 5, no. 9, pp. 4652–4692, 2013.

[20] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-6, no. 6, pp. 721–741, Nov. 1984.

[21] M. Lhuillier and L. Quan, "Quasi-dense reconstruction from image sequence," in *Proc. Eur. Conf. Comput. Vis*, 2002, pp. 125–139.

[22] D. Anguelov *et al.*, "Discriminative learning of Markov random fields for segmentation of 3D scan data," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2005, vol. 2, pp. 169–176.

[23] D. Munoz, J. A. Bagnell, N. Vandapel, and M. Hebert, "Contextual classification with functional max-margin Markov networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 975–982.

[24] D. Munoz, N. Vandapel, and M. Hebert, "Onboard contextual classification of 3-D point clouds with learned high-order Markov random fields," in *Proc. IEEE Conf. Robot. Autom.*, 2009, pp. 2009–2016.

[25] A. Nguyen and B. Le, "Contextual labeling 3D point clouds with conditional random fields," in *Intell. Inf. Database Syst.*, 2014, pp. 581–590.

[26] J. Niemeyer, F. Rottensteiner, and U. Soergel, "Contextual classification of LiDAR data and building object detection in urban areas," *ISPRS J. Photogramm. Remote Sens.*, vol. 87, pp. 152–165, Jan. 2014.

[27] M. Najafi, S. T. Namin, M. Salzmann, and L. Petersson, "Non-associative higher-order Markov networks for point cloud classification," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 500–515.

[28] P. Kohli, M. P. Kumar, and P. H. Torr, "P3 & beyond: Solving energies with higher order cliques," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2007, pp. 1–8.

[29] P. Kohli, L. Ladický, and P. H. Torr, "Robust higher order potentials for enforcing label consistency," *Int. J. Comput. Vis.*, vol. 82, no. 3, pp. 302–324, May 2009.

[30] A. Golovinskiy, V. G. Kim, and T. Funkhouser, "Shape-based recognition of 3D point clouds in urban environments," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2009, pp. 2154–2161.

[31] B. Yang and Z. Dong, "A shape-based segmentation method for mobile laser scanning point clouds," *ISPRS J. Photogramm. Remote Sens.*, vol. 81, pp. 19–30, Jul. 2013.

[32] H. Wang *et al.*, "Object detection in terrestrial laser scanning point clouds based on Hough forest," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 10, pp. 1807–1811, Oct. 2014.

[33] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (FPFH) for 3D registration," in *Proc. IEEE Conf. Robot. Autom.*, 2009, pp. 3212–3217.

[34] E. Wahl, U. Hillenbrand, and G. Hirzinger, "Surflet-pair-relation histograms: A statistical 3D-shape representation for rapid classification," in *Proc. Int. Conf. 3-D Digit. Imag. Model.*, 2003, pp. 474–481.

[35] C. Barnes, E. Shechtman, A. Finkelstein, and D. Goldman, "Patchmatch: A randomized correspondence algorithm for structural image editing," *ACM Trans. Graph.*, vol. 28, no. 3, pp. 1–11, 2009.

[36] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 11, pp. 1222–1239, Nov. 2001.

[37] RIEGL VMX-450 Datasheet. Accessed 2-May-2015. [Online]. Available: http://www.riegl.com/nc/products/mobile-scanning/produktdetail/product/scannersystem/10/

[38] A. Liaw and M. Wiener, "Classification and regression by random forest," *R Newsletter*, vol. 2, no. 3, pp. 18–22, 2002.

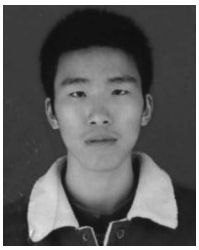
[39] C. J. Burges, "A tutorial on support vector machines for pattern recognition," *Knowl. Discov. Data Mining*, vol. 2, no. 2, pp. 121–167, Jun. 1998.

[40] C.-C. Chang and C.-J. Lin, LIBSVM—A library for support vector machines. [Online]. Available: http://www.csie.ntu.edu.tw/~cjlin/libsvm/

[41] T. Kanungo *et al.*, "An efficient k-means clustering algorithm: Analysis and implementation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 881–892, Jul. 2002.

[42] K. Park and S. Gould, "On learning higher-order consistency potentials for multi-class pixel labeling," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2012, pp. 202–215.

[43] L. Ladicky, C. Russell, P. Kohli, and P. H. Torr, "Graph cut based inference with co-occurrence statistics," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 239–253.

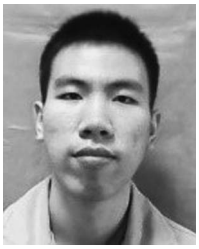


Huan Luo received the B.Sc. degree in software engineering from Nanchang University, Nanchang, China, in 2009. He is currently working toward the Ph.D. degree in the Fujian Key Laboratory of Sensing and Computing for Smart City, School of Information Science and Engineering, Xiamen University, Xiamen, China. His current research interests include computer vision, machine learning, and mobile LiDAR point cloud data processing.

Cheng Wang (M'11) received the Ph.D. degree in information and communication engineering from the National University of Defense Technology, Changsha, China, in 2002.

He is currently a Professor with and the Associate Dean of the School of Information Science and Technology, Xiamen University, Xiamen, China. He has authored more than 80 papers. His research interests include remote sensing image processing, mobile LiDAR data analysis, and multisensor fusion. Dr. Wang is a member of the SPIE and the IEEE Geoscience and Remote Sensing Society and a Council Member of the China Society of Image and Graphics. He is the Cochair of ISPRS WG I/3.

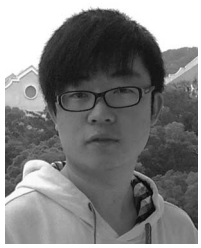
Chenglu Wen (M'14) received the Ph.D. degree in mechanical engineering from China Agricultural University, Beijing, China, in 2009. She is currently an Assistant Professor with the Fujian Key Laboratory of Sensing and Computing for Smart City, School of Information Science and Engineering, Xiamen University, Xiamen, China. She has coauthored more than 30 research papers published in refereed journals and proceedings. Her current research interests include machine vision, machine learning, and point cloud data processing. She is the Secretary of the ISPRS WG I/3 on Multi-Platform Multi-Sensor System Calibration (2012–2016).



Zhipeng Cai received the B.Sc. degree in computer science from Xiamen University, Xiamen, China, in 2013, where he is currently working toward the M.Sc. degree in the Fujian Key Laboratory of Sensing and Computing for Smart City, School of Information Science and Engineering. His current research interests include computer vision, machine learning, and LiDAR point cloud processing.

Ziyi Chen received the B.S. degree in computer science from Xiamen University, Xiamen, China, in 2010, where he is currently working toward the Ph.D. degree in the Department of Communication Engineering. His current research interests include computer vision, machine learning, and remote sensing image processing.

Hanyun Wang (M'13) received the M.Sc. degree in information and communication engineering from the National University of Defense Technology, Changsha, China, in 2010, where he is currently working toward the Ph.D. degree in the School of Electronic Science and Engineering. His research interests include computer vision, machine learning, and LiDAR point cloud processing.



Yongtao Yu received the B.S. degree in computer science and technology from Xiamen University, Xiamen, China, in 2010, where he is currently working toward the Ph.D. degree in computer science and technology in the Department of Computer Science. He has coauthored more than 20 research papers published in refereed journals and proceedings. His current research interests include pattern recognition, computer vision, machine learning, mobile laser scanning, and information extraction from 3-D point clouds.

Jonathan Li (M'00–SM'11) received the Ph.D. degree in geomatics engineering from the University of Cape Town, Cape Town, South Africa.

He is currently a Professor with the Key Laboratory of Underwater Acoustic Communication and Marine Information Technology of the Ministry of Education, School of Information Science and Engineering, Xiamen University, Xiamen, China. He is also a Professor with and the Head of the GeoSTARS Lab, Faculty of Environment, University of Waterloo, Waterloo, ON, Canada. He has coauthored more than 300 publications, more than 100 of which were published in refereed journals, including IEEE TGRS, IEEE TITS, IEEE GRSL, ISPRS JPRS, IJRS, PE&RS, and RSE. His current research interests include information extraction from mobile LiDAR point clouds and from Earth observation images.

Dr. Li is the Chair of the ISPRS WG I/Va on Mobile Scanning and Imaging Systems (2012–2016), the Vice Chair of the ICA Commission on Mapping from Remote Sensor Imagery (2011–2015), and the Vice Chair of the FIG Commission on Hydrography (2015–2018).