2013 IEEE/RSJ International Conference on
Intelligent Robots and Systems (IROS)
November 3-7, 2013. Tokyo, Japan

Nonparametric Semantic Segmentation for 3D Street Scenes

Hu He and Ben Upcroft

Abstract— **In this paper we propose a method to generate a large scale and accurate dense 3D semantic map of street scenes. A dense 3D semantic model of the environment can significantly improve a number of robotic applications such as autonomous driving, navigation or localisation. Instead of using offline trained classifiers for semantic segmentation, our approach employs a data-driven, nonparametric method to parse scenes which easily scale to a large environment and generalise to different scenes.**

We use stereo image pairs collected from cameras mounted on a moving car to produce dense depth maps which are combined into a global 3D reconstruction using camera poses from stereo visual odometry. Simultaneously, 2D automatic semantic segmentation using a nonparametric scene parsing method is fused into the 3D model. Furthermore, the resultant 3D semantic model is improved with the consideration of moving objects in the scene. We demonstrate our method on the publicly available KITTI dataset and evaluate the performance against manually generated ground truth.

I. INTRODUCTION

Recently, there has been an increased interest in developing intelligent autonomous systems for a wide range of applications such as autonomous driving, robot navigation or environment exploration. To enable autonomous system operation in large scale and dynamic environments, these systems need the ability to understand the environment geometrically and semantically.

Considerable effort has been focussed towards geometrically modelling the environment with a map consisting of sparse or dense point cloud for accurate localisation [1]–[4]. However, none of these approaches consider the semantic aspects of the environment which can provide helpful information for the geometric model, *e.g.*, buildings, cars and pedestrians should be on top of a road and under the sky. Moreover, pedestrians and cars are likely to be moving objects which are not suitable for generating a static navigation map compared to buildings or trees. Semantic scene parsing (*i.e.*, label each pixel in a image into a semantic class) has been actively researched by the computer vision community [5]–[8], but mainly focused on 2D images. There is relatively little work on combining semantic and geometric representations of the environment.

Douillard *et al.* proposed a sparse semantic map representation of the environment using a laser and camera [9]. However, such a sparse semantic map has difficulty obtaining the boundary between objects, resulting in inaccurate classification. Instead of using multiple sensors, we aim for a vision only system to alleviate the cost and complexity.

The authors are with School of Electrical Engineering and Computer Science, Queensland University of Technology, Gardens Point, Queensland, Australia. {h2.he,ben.upcroft}@qut.edu.au

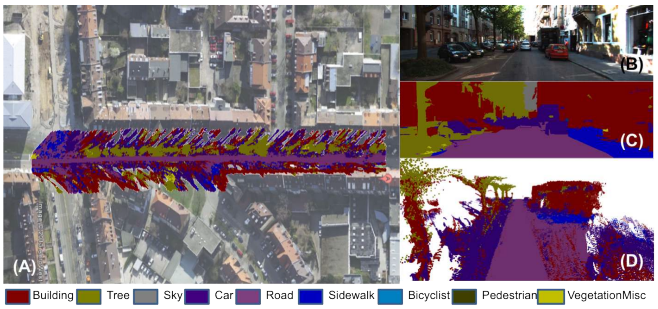


Fig. 1: 3D semantic model. This figure shows some sample outputs of our method (best viewed in color). (A) 3D semantic map overlaid on Google Earth map; (B) Left view of a stereo image pair; (C) 2D semantic segmentation results; (D) 3D semantic model.

[10] presented a method for 3D semantic map generation, but in that work they assume the environment is static. In addition, they apply a parametric method for semantic image segmentation which requires offline training that needs to be updated whenever the environment changes. The method presented here employs a data-driven nonparametric semantic segmentation without an offline training procedure which can easily scale to a large scale environment.

In this paper, we propose a method of joint 3D reconstruction with nonparametric semantic segmentation to model the environment as a 3D semantic occupancy map. Additionally, moving objects are taken into account using estimated camera poses from stereo visual odometry (§III-D). Finally, the proposed method is qualitatively and quantitatively evaluated on the **KITTI** dataset [11]. Some intermediate and final outputs of our method are shown in Fig. 1. Our main contributions can be summarised as follows:

- We apply a data-driven nonparametric method for semantic image segmentation which does not require any offline training procedure. Therefore our method can easily scale to a large dataset.
- We represent the 3D semantic map as an octree structure which introduces efficiency in terms of time and space.
- We take moving objects in the scene into account using estimated camera poses to improve the 3D semantic map.
- The production of a semantic segmentation ground truth is time consuming. We make our hand labelled ground truth of different street scene sequences publicly available[1].

The rest of this paper is organised as follows. The follow-

[1] https://wiki.qut.edu.au/display/cyphy/Hu+He

ing section addresses related work. Section III describes the whole framework of the proposed method in details. Qualitative and quantitative experimental evaluation are discussed in Section IV. Finally, Section V draws the conclusion and states future work.

II. RELATED WORK

Recently, there exists a large amount of work from the computer vision community focused on nonparametric, data-driven modelling for scene inference which does not require offline training procedures [12], [13]. These nonparametric methods need a pre-built database that contains raw data and labelled data, and then parse the input images using a recognition-by-matching method. Specifically, the visual objects in an input image are matched with the images in a database using appearance similarity (*e.g.*, SIFT flow [14]). As the matched images in the database are annotated, the labels of the images in the database can be transferred to the input image if the match is semantically meaningful (*i.e.*, building corresponds to building, tree corresponds to tree). Furthermore, these initial labels are fused into Markov Random Fields (MRFs) or Conditional Random Fields (CRFs) to improve the labelling for each pixel or superpixel (*i.e.*, pixel sets with homogeneous attributes). In this paper, we extend the above 2D nonparametric semantic segmentation to a 3D space.

The most related literature to our work are [10], [15], [16]. [15] employs 3D data from a laser or RGB-D sensor to create a 3D probabilistic occupancy map for the environment without semantic labelling. In [16], nonparametric models are applied to image and video parsing. However, they do not extend their method to the 3D space which is key for most robotic applications. Very recently, Sengupta *et al.* [10] proposed a 3D semantic model of street scenes. However, they employ parametric models for 2D image semantic segmentation which requires both offline training and large amount of training time especially for a large training dataset. Additionally, they apply high-order CRFs for semantic label inference. We argue that a simple potential term, modelled by sufficiently expressive observations, is comparable with the strong prior potential term (*i.e.*, high-order potential) in random fields. Therefore we apply second-order MRFs with a simple term for semantic inference. Furthermore, we also take into account moving objects in the scene and evaluate our method over multiple **KITTI** datasets.

III. 3D SEMANTIC MODEL FOR SCENE UNDERSTANDING

In this section we introduce our proposed method for 3D semantic occupancy map creation. As shown in Fig. 2, our approach has three parallel stages: estimate the semantic label of each pixel in the left view of each stereo image pair; camera pose for each stereo image pair; and dense depth map for each left view of stereo image pairs. Then a 3D occupancy map is constructed based on the reconstructed point clouds and a semantic label from a previous 2D nonparametric image segmentation is obtained. Finally, the 3D semantic model is updated based on camera trajectory estimated from stereo visual odometry, taking into account moving objects along the trajectory. In the following, we explain each stage in detail.

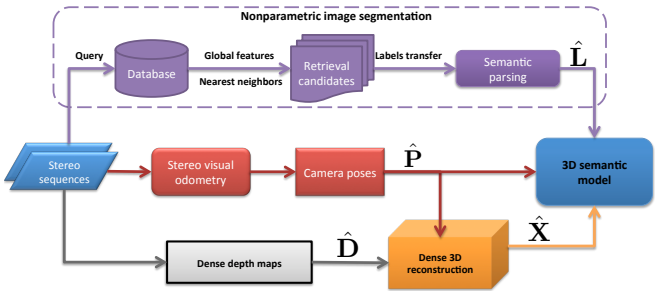


Fig. 2: System overview. Given rectified stereo image pairs, semantic labels ($\hat{\mathbf{L}}$) are inferred by nonparametric segmentation method. Camera poses ($\hat{\mathbf{P}}$) are estimated using stereo visual odometry. Dense 3D reconstruction ($\hat{\mathbf{X}}$) is computed using dense depth maps ($\hat{\mathbf{D}}$) and camera poses. Based on these information, the final 3D semantic model can be obtained.

A. Nonparametric 2D Semantic Segmentation

As in [17], the pixels in an image and their corresponding semantic labels are represented by a MRF which is defined on a graph $\mathsf{G} = <\mathsf{V},\mathsf{E}>$ consisting of N nodes (*i.e.*, $|\mathsf{V}| = N$), where each node $v_i \in \mathsf{V}$ represents the latent random variable associated with the pixel i in the image and each edge $e_{ij} \in \mathsf{E}$ represents the relationship of two neighbouring nodes, *i.e.*, v_i and v_j in the graph.

The segmentation problem for an image can be considered as a labelling problem in which every pixel should be assigned a unique label l. In this paper, $l = \{$*Building, Car, Sky, Tree, Sidewalk, Road, Bicyclist, Pedestrian, Vegetation-Misc*$\}$. Therefore, the solution $\mathbf{L} = (l_1, \cdots, l_i, \cdots, l_N)$ can be achieved by maximising $p(\mathbf{L})$ given by a Gibbs distribution of the following form:

$$p(\mathbf{L}) = \frac{1}{Z} exp(-\sum E(\mathbf{L})) \tag{1}$$

where Z is a normalisation constant.

Now the maximisation of $p(\mathbf{L})$ is equivalent to the minimisation of an energy function $E(\mathbf{L})$. Following Bayes rule, we can formulate the energy function $E(\mathbf{L})$ as follows:

$$E(\mathbf{L}) = \sum_{i \in \mathsf{V}} \psi_i(l_i) + \lambda \sum_{(i,j) \in \mathsf{E}} \psi_{ij}(l_i, l_j) \tag{2}$$

where $\psi_i(l_i)$ is the unary potential (*i.e.*, likelihood energy) encoding the cost when the label of the pixel i is l_i, and $\psi_{ij}(l_i, l_j)$ is the pairwise potential (*i.e.*, prior energy) representing the cost when the label for adjacent pixels i and j are l_i and l_j, respectively. λ indicates the relative importance of the likelihood energy versus the prior energy. We use $\lambda = 1$ in our experiments.

Unary potential: Referring to the TextonBoost algorithm [8] which combines classifiers with different feature representations to model the unary potential term in (2), we

also apply a multi-feature representation to model this energy term. However, we employ a nonparametric method to transfer the likelihood probability of labelled images in the database to the query image rather than use pre-trained classifiers described in [8]. More specifically, we build up a database which contains raw images and corresponding manually labeled ground truth. Like in [16], we use *gist* [18], *color histogram* and a *visual words dictionary* [19] as global features to represent each image in the database. And then we apply the mean shift algorithm to generate oversegmentation on each image in the database (each segment denotes a *super pixel*). Local features like SIFT, location in image coordinates, size of the super pixel in pixels, color of each super pixel are extracted and concatenated to represent each super pixel [16]. Each of the images in the database have been labelled, giving each super pixel a corresponding semantic label. Regarding a query image, global features for the entire image and local features for the generated *super pixels* are extracted. Then images from the database are ranked based on the similarity of global features to the query image. We choose the 30 top-ranked images from the database as the nearest neighbours for modelling likelihood probability of each super pixel in the query image to be each semantic label as follows:

$$\psi_i(l_i) = \psi_i(l_{sp_i}) = -log \prod_{\mathbf{sp} \in \mathbf{I}} w_i p(\mathbf{sp}|l) \tag{3}$$

where sp_i represents the super pixel containing pixel i in the query image, **sp** denotes super pixels from nearest neighbours **I**, and $l \in \mathbf{L}$ is the semantic label. w_i is the normalised distance between super pixels in the query image and nearest neighbours from database. For more details on the nonparametric method, the reader is encouraged to refer to [12], [16].

Pairwise potentials: The 8-neighbourhood smoothness prior term ϕ_{ij} modelling the probabilities of label co-occurrence (*i.e.*, encouraging the adjacent pixels take the same label). We model this term using a contrast sensitive Potts Model [20].

$$\psi_{ij}(l_i, l_j) = |l_i - l_j| exp(-\frac{||C_i - C_j||^2}{2\sigma^2}) \tag{4}$$

where C_i denotes the RGB value of a pixel i and $||C_i - C_j||^2$ is the Euclidean norm of the intensity difference. σ is the average intensity difference between neighbouring pixels in the image, which can be estimated as pixel noise introduced by the camera. This smoothness term favours the object boundary where neighbouring pixels have large contrast.

This MRF can be solved using the standard graph cut algorithm [17], [20].

B. Stereo Visual Odometry

As camera pose estimation is not the primary focus of this work, we apply the modified stereo visual odometry system described in [21]. The input data are rectified stereo image pairs from calibrated cameras. Stereo matching turns into a 1D search (*i.e.*, horizontal correspondence) which is quite efficient. With respect to the feature matching over time, conventional camera resectioning [22] is applied to estimate camera poses over time with a fixed calibration assumption. In order to reject the incorrect matching (*i.e.*, outliers) due to lack of texture or image noise, we ensure the visibility of a detected feature exists for at least three consecutive frames over time for the stereo pair. Additionally, features from dynamic objects are discarded using epipolar geometric constraints. One example result of camera pose estimation is shown in Fig. 3. As the dataset (**KITTI Odometry sequence 15**) does not have ground truth, we illustrate the camera trajectory overlaid on Google map image qualitatively.

Fig. 3: Camera trajectory computed by stereo visual odometry is registered manually to the Google Map.

C. Dense 3D Reconstruction

For dense 3D reconstruction, we firstly generate dense depth maps for each stereo pair. Specifically, we apply the efficient stereo matching algorithm proposed in [23] to compute a disparity map between stereo images, and then filter out extreme disparity values using a median filter with a 3×3 patch window for an image of resolution 1241×376. As the stereo camera is calibrated, we use (5) to compute dense point clouds for each pixel with a valid disparity expressed in the left camera coordinate.

$$X_i = (x_i - c_x)B/d_i \tag{5a}$$
$$Y_i = (y_i - c_y)B/d_i \tag{5b}$$
$$Z_i = fB/d_i \tag{5c}$$

where (X_i, Y_i, Z_i) is the 3D point expressed in the left camera frame corresponding to the pixel i with valid disparity d_i at (x_i, y_i) in image space. B and f denote baseline and focal length, while (c_x, c_y) represents the principal point in image space derived from stereo calibration.

Secondly, a camera viewing volume (*i.e.*, viewing frustum) is clipped into a [0.5m 20m] depth range and then transferred into global coordinates (origin is usually chosen as the pose of the initial camera). The 3D volume is divided into voxels with 0.2m resolution using an octree [15]. Each minimum voxel denotes the leaf node in the octree and is derived from parent nodes (see Fig. 4). Then we compute the average centre of 3D points from the same voxel to

represent the location of that voxel in global coordinates. The semantic labels for these inside 3D points can be obtained using estimated camera poses (§III-B) and 2D semantic segmentations (§III-A). Finally, the semantic label for that voxel is taken from the most frequent semantic label of the inside 3D points. Note that the leaf nodes can be pruned if all eight leaf nodes take the same semantic labels, their parent node will take that semantic label and represent them. The advantage of this representation is to decrease the data size requirements and also increase the processing efficiency.

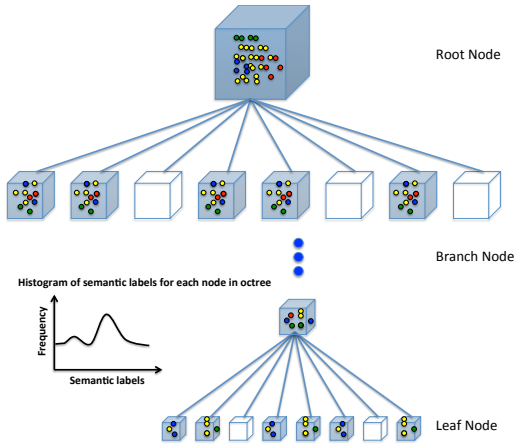


Fig. 4: 3D semantic model is organised as an octree. Different node in the tree has different metric resolution. We take the resolution of leaf nodes as 0.2m in this paper. A different color of 3D points in each voxel denotes the different semantic label. The voxel semantic label is determined based on the maximum of the semantic label histogram of inside 3D points. The empty voxel represents the free space while the shaded voxel denotes the occupied voxel (best viewed in color).

Finally, we update the volume using the camera pose from stereo visual odometry. In order to increase memory efficiency, we write the volume behind the cameras to disk to deal with a larger dataset. With respect to the occupancy estimation, we label the voxel as occupied if that voxel contains more than 5 reconstructed 3D points and as free space otherwise. Note that the above method assumes the environment is static. Moving objects might introduce duplicate points for the same object in the 3D reconstruction (see Fig. 6(a)). A simple moving object filter will be described in Section III-D to address this issue.

D. 3D Semantic Model

In this section we introduce the way we generate our final 3D semantic model. Once the semantic segmentation for each image is obtained, we use the camera projection matrix to project the color of the semantic label to the reconstructed 3D points. As previously mentioned, each occupied voxel would contain multiple 3D points with different semantic labels (see Fig. 4). We compute the label histogram in each occupied voxel and choose the most frequent label as the voxel semantic label.

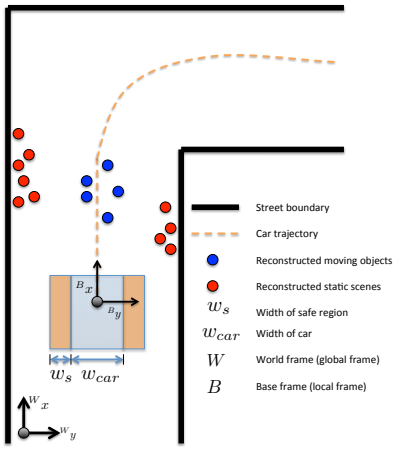
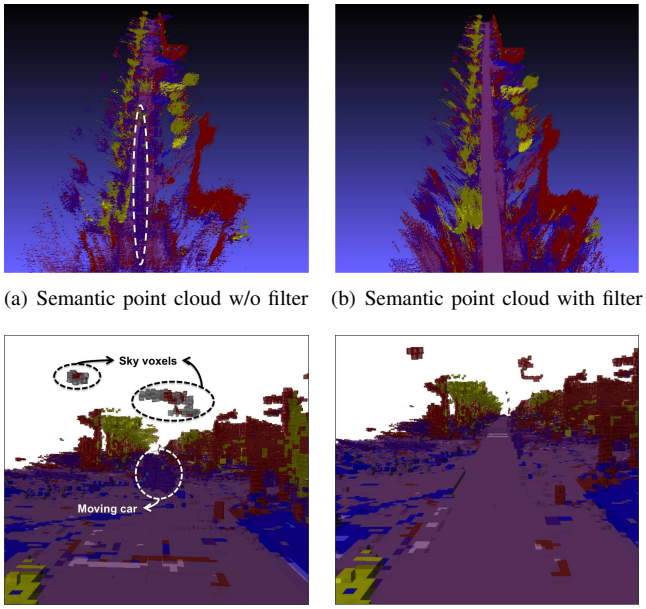


Fig. 5: Demonstration of our moving objects filter (best viewed in color). Blue dots represent the incorrect reconstruction due to moving objects along the camera trajectory. Blue dots will be filtered out and red dots will be kept.



(a) Semantic point cloud w/o filter (b) Semantic point cloud with filter

(c) Semantic volume w/o filter (d) Semantic volume with filter

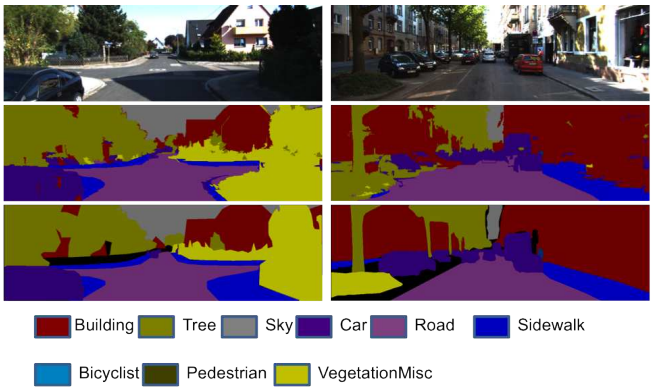
Fig. 6: Qualitative comparison between 3D semantic model without and with a moving objects filter (best viewed in color). Regions of interest are highlighted by dash ellipses.

Due to moving objects and textureless areas (*e.g.*, *sky*) in the scene, the 3D model might contain incorrect reconstructions. As shown in Fig. 6(a) and 6(c), moving objects are duplicated in the 3D model and parts of the sky are reconstructed. As we know the semantic label for the occupied voxel in the 3D model, we can correct the occupied voxel with a *sky* label as free space (see Fig. 6(d)).

With respect to the error introduced by moving objects, we employ a simple yet effective method to filter the 3D map. Once we obtain the 3D semantic map and camera trajectory, we argue that the region where the car can drive through should be free 3D space. Therefore, the region covered by

the camera trajectory is traversable. In Fig. 5, we know the width (w_{car}) of the car on which the stereo rig is mounted. In addition, there are free regions (w_s) between the car and other obstacles for safety purposes. Any occupied voxels within this bounding box defined by w_{car} are removed if their semantic labels are not *road*. As expected, it causes a significant number of holes in the 3D model, however, we know they are likely to be *road*. We use the geometric information from the remaining voxels (most of them should be *road*) within the bounding box defined by $w_{car} + w_s$ to generate new *road* voxels to fill the holes. In particular, we adjust the $w_s \in$[0.3m 0.5m] (reasonable safe distance between cars) to achieve the smallest standard deviation along z-axis (*i.e.*, altitude above the sea level). The resultant 3D semantic map is shown in Fig. 6(b) and 6(d).

(a) Sample images and ground truth in our database



(b) Semantic image segmentation

Fig. 7: 7(a) Left: raw images; Right: ground truth. Note that the object with the same class has different appearance in our database highlighted in red dash ellipse; 7(b) Top: input images; Middle: semantic segmentation results; Bottom: corresponding ground truth. (best viewed in color)

IV. EXPERIMENTAL EVALUATION

In this section, we describe the dataset used and a qualitative and quantitative evaluation of our system. Additionally, we compare our results with that reported in [10].

A. Datasets

We evaluate our method on two publicly available **KITTI** datasets: **2011_09_26_drive_0104** and **Odometry sequence 15**. Both datasets contain rectified stereo pairs with associated 3D ground truth data obtained by a Velodyne HDL-64E laser scanner which is calibrated with respect to the stereo camera. These datasets involved common objects such as *pedestrian*, *bicyclist*, *car*, *tree* or *building* in urban, residential and campus like environment. **2011_09_26_drive_0104** consists of 312 image pairs at 1242×375 pixel resolution over a driving distance of about 252m. We manually label 5 images for this dataset. **Odometry sequence 15** contains 1901 stereo pairs with a resolution of 1241×376 covering a track of around 1.5km, and we generate 7 ground truth images for this dataset. We label the scene into 9 semantic classes, *i.e.*, *Building, Car, Sky, Tree, Sidewalk, Road, Bicyclist, Pedestrian, VegetationMisc*. In addition, we manually annotated another 39 images from other **KITTI** datasets for our database setup. Note that the 12 ground truth images from **2011_09_26_drive_0104** and **Odometry sequence 15** datasets were used for testing. These datasets are quite challenging, and even objects of the same class in the scene have significantly different appearance (see Fig. 7(a)).

In our current implementation, 3D reconstructions can run up to 4∼5Hz, and 2D semantic segmentation of query image takes around 30s. However, most processing time is consumed by feature extraction and matching which can be parallelised using a GPU implementation.

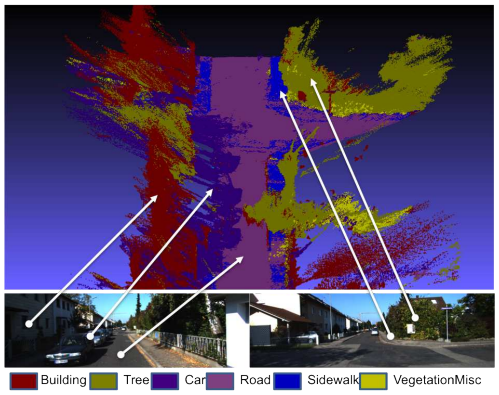
B. Qualitative Results

Fig. 7 shows 2D semantic segmentation results from the nonparametric image parsing model. The top row is the sample images from the dataset. The middle row shows the results from the nonparametric model. By comparing with ground truth shown in the bottom row, we can achieve quite plausible semantic segmentation results, especially for classes such as *Building*, *Road* or *Tree*. We also notice that *Tree* and *VegetationMisc* are partially mislabeled due to similar appearance and location. Additionally, shadow causes the *VegetationMisc* label to bleed into the *Sidewalk* label (see left column). These effects attribute to the quantitative results in Table I. Qualitative correspondences highlighted by white arrows between the 2D images and 3D semantic model are shown in Fig. 8(a). A large 3D semantic map has been created using **Odometry sequence 15** and overlaid on the corresponding Google Earth map as illustrated in Fig. 8. This 3D semantic map is more than two times larger than that addressed in [10]. More qualitative results are shown in a supplementary video with this paper.

C. Quantitative Results

For quantitative evaluation, we firstly evaluate the geometric accuracy of our 3D model using the Velodyne based ground truth and then the semantic accuracy using our manually labelled ground truth.

Geometric accuracy evaluation: By following [24], we using laser measures from Velodyne to evaluate the 3D model. Specifically, we project the 3D model and the corresponding ground truth 3D laser data back into a 2D image space. Using the inverse form of (5), *i.e.*, computing disparity based on 3D points and camera information, we can generate disparity maps for our 3D model and 3D laser data. For each ground



(a) Closeup view of the 3D semantic model

(b) Large scale 3D semantic map

Fig. 8: 8(a): White arrows show the correspondent objects between 3D semantic model and 2D images; 8(b): A 3D semantic map with a 1.5km track overlaid on Google Earth map manually.

truth, we compute the ratio between the number of pixels that satisfy $|d_i - d_i^g| \geq \delta$ and the number of valid projection of laser 3D data, where d_i^g is the ground truth disparity computed from laser 3D data corresponding to the disparity d_i generated from our 3D model. The error tolerance δ ranges from 1 to 8 pixels. Then the ratios are averaged over all the ground truth data (see Fig. 9). Particularly, the average incorrect pixel ratio is around 10% (*i.e.*, 90% accuracy) given an error tolerance of 5 pixels.

Semantic accuracy evaluation: We use the evaluation measures defined in [5], [10] to compute *per-class Recall (R), Average Recall (AR), Global Recall (GR)* and *per-class Intersection vs Union (IU)* for 2D semantic image segmentation, the 3D model without a filter and the 3D model with a filter. *GR* evaluates the overall ratio of correct labelling, and *AG* denotes the average recall score of the per-class measures. We use the selected camera poses to project our 3D semantic model (*i.e.*, without a filter and with a filter) back to image views which have manually labelled ground truth (see Fig. 10). Note that we set the depth range of the camera viewing frustum as [0.5m 20m]. Thus, any structures beyond this range are ignored during our evaluation.

As shown in Table I, the frequent and dominant classes in street scenes like *Road, Building* or *Car* achieve reasonably high accuracy in the 2D semantic image segmentation and 3D semantic model. There are few images containing *Pedestrian* and *Bicyclist* labels in our current database (only 39 images), therefore the accuracy for these classes is quite low compared with the other classes. Due to the error in camera pose estimation and 3D reconstruction, 2D semantic image segmentation always outperforms the 3D semantic model. We can also see that our 3D semantic model with a filter obtains better performance than that without a filter. As expected, most improvements occur in the *Road* class on which the filter takes effect. Additionally, we compare our results with that in [10]. They consider *Tree* and *Vegetation* as the same class, and evaluate their results by ignoring several classes due to insufficient training data. In order to make a fair comparison, we compute the average score (marked by †) using the common classes between our experiment and [10]. Note that we also compute the average score for all the classes parsed by our model. The performance of some classes (*e.g., Building, Car, Road*) is comparable with [10]. While our performance on classes like *Tree, VegetationMisc* or *Sidewalk* (named as *Pavement* in [10]) is inferior, they use a more sophisticated graphical model (*i.e.*, high order CRFs) and offline training with the images from the same sequence. However, we consider *Tree* and *Vegetation* as different classes (the inter-class similarity introduces more error to our model). Additionally, the images in our database are from different sequences rather than the test sequence.

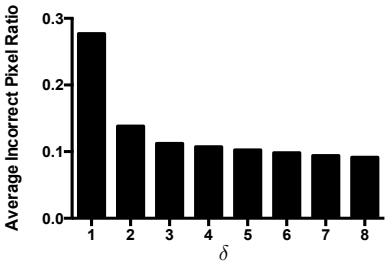


Fig. 9: Dense 3D reconstruction evaluation. δ is the error tolerance.

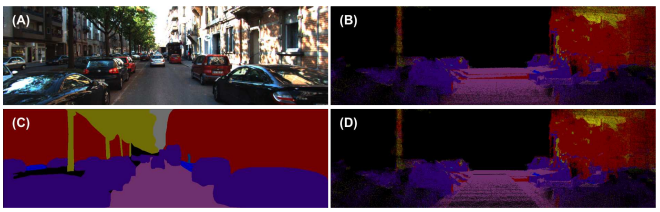


Fig. 10: 3D semantic model evaluation (best viewed in color). (A) the input image; (B) corresponding projected view from 3D semantic model without a filter; (C) ground truth; (D) corresponding projected view from 3D semantic model with a filter.

V. CONCLUSIONS AND FUTURE WORK

We have presented a method for 3D street scene understanding using nonparametric semantic segmentation and dense 3D reconstruction. We also take into account moving objects in the scene to improve the 3D semantic occupancy map. The evaluation on several challenging **KITTI** street

Method	Building	Sky	Car	Road	Pedestrian	Sidewalk	Bicyclist	Tree	Vegetation	**Average†**	**Average**	**Global**
Recall												
Image segmentation	93.54	97.2	96.82	97.16	29.58	79.88	0.0	94.65	29.04	91.85	68.65	92.77
Image segmentation [10]	97.0	-	93.9	98.3	-	91.3	-	-	-	81.68	-	88.4
Semantic model w/o filter	80.19	0.0	81.31	81.69	0.0	30.25	0.0	48.3	0.9	68.36	35.85	78.48
Semantic model with filter	80.19	0.0	81.31	88.46	0.0	30.25	0.0	48.3	0.9	70.05	36.6	79.64
Semantic model [10]	96.1	-	88.5	97.8	-	86.5	-	-	-	77.15	-	85
Intersection vs Union												
Image segmentation	90.63	80.15	91.22	93.85	29.58	71.54	0.0	87.67	23.52	86.81	63.13	
Image segmentation [10]	86.1	-	78.0	94.3	-	73.4	-	-	-	71.65	-	
Semantic model w/o filter	71.36	0.0	68.61	75.87	0.0	27.43	0.0	10.57	0.9	60.82	28.3	
Semantic model with filter	72.93	0.0	69.37	80.98	0.0	27.43	0.0	10.57	0.9	62.68	29.13	
Semantic model [10]	83.8	-	63.5	96.3	-	68.4	-	-	-	65.7	-	

TABLE I: Quantitative results on the **KITTI** dataset. 3D semantic with a filter outperforms that without a filter. (†) indicates the score is computed using the common classes between our experiment and [10].

scene datasets shows the promise of our method for 3D scene understanding.

In future work, we plan to investigate the employment of motion features for improving dynamic scene parsing. In addition, we are also interested in exploring the interplay between 2D and 3D information. Finally, we would like to incorporate the 3D semantic map into SLAM framework towards a semantic SLAM method.

ACKNOWLEDGMENTS

The authors would like to thank Inkyu Sa and Alex Bewley for help in annotating the dataset. The authors also would like to thank Michael Warren, Gordon Wyeth and Peter Corke for the helpful discussions. The authors also thank Derek Hoiem for providing the ground truth annotation tool. Hu He is funded by a Chinese Scholarship Council (CSC) scholarship.

REFERENCES

[1] C. Hane, C. Zach, J. Lim, A. Ranganathan, and M. Pollefeys, "Stereo depth map fusion for robot navigation," in *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*. IEEE, 2011, pp. 1618–1625.

[2] S. Se, D. Lowe, and J. Little, "Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks," *The international Journal of robotics Research*, vol. 21, no. 8, pp. 735–758, 2002.

[3] K. Konolige, J. Bowman, J. Chen, P. Mihelich, M. Calonder, V. Lepetit, and P. Fua, "View-based maps," *The International Journal of Robotics Research*, vol. 29, no. 8, pp. 941–957, 2010.

[4] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse, "MonoSLAM: Real-time single camera SLAM," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, no. 6, pp. 1052–1067, 2007.

[5] L. Ladicky, C. Russell, P. Kohli, and P. H. Torr, "Associative hierarchical CRFs for object class image segmentation," in *Computer Vision, 2009 IEEE 12th International Conference on*. IEEE, 2009, pp. 739–746.

[6] G. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla, "Segmentation and recognition using structure from motion point clouds," *Computer Vision–ECCV 2008*, pp. 44–57, 2008.

[7] P. Sturgess, K. Alahari, L. Ladicky, and P. Torr, "Combining appearance and structure from motion features for road scene understanding," *BMVC*, 2009.

[8] J. Shotton, J. Winn, C. Rother, and A. Criminisi, "Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation," *Computer Vision–ECCV 2006*, pp. 1–15, 2006.

[9] B. Douillard, D. Fox, F. Ramos, and H. Durrant-Whyte, "Classification and semantic mapping of urban environments," *The International Journal of Robotics Research*, vol. 30, no. 1, pp. 5–32, 2011.

[10] S. Sengupta, E. Greveson, A. Shahrokni, and P. H. Torr, "Urban 3d semantic modelling using stereo vision," in *International Conference on Robotics and Automation (ICRA)*, Karlsruhe, Germany, May 2013.

[11] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving?" in *Computer Vision and Pattern Recognition (CVPR*, Providence, USA, June 2012.

[12] C. Liu, J. Yuen, and A. Torralba, "Nonparametric scene parsing via label transfer," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 33, no. 12, pp. 2368–2382, 2011.

[13] A. Torralba, R. Fergus, and W. T. Freeman, "80 million tiny images: A large data set for nonparametric object and scene recognition," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 30, no. 11, pp. 1958–1970, 2008.

[14] C. Liu, J. Yuen, A. Torralba, J. Sivic, and W. Freeman, "SIFT flow: Dense correspondence across different scenes," *Computer Vision–ECCV 2008*, pp. 28–42, 2008.

[15] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "Octomap: an efficient probabilistic 3d mapping framework based on octrees," *Autonomous Robots*, pp. 1–18, 2013.

[16] J. Tighe and S. Lazebnik, "Superparsing: scalable nonparametric image parsing with superpixels," *Computer Vision–ECCV 2010*, pp. 352–365, 2010.

[17] Y. Boykov and M. Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in nd images," in *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, 2001, pp. 105–112.

[18] A. Torralba, "Contextual priming for object detection," *International Journal of Computer Vision*, vol. 53, no. 2, pp. 169–191, 2003.

[19] M. Cummins, "Probabilistic localization and mapping in appearance space," Ph.D. dissertation, University of Oxford, 2009.

[20] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, pp. 1124–1137, 2004.

[21] B. Kitt, A. Geiger, and H. Lategahn, "Visual odometry based on stereo image sequences with RANSAC-based outlier rejection scheme," in *IEEE Intelligent Vehicles Symposium*, San Diego, USA, June 2010.

[22] D. Scaramuzza and F. Fraundorfer, "Visual odometry [tutorial]," *Robotics & Automation Magazine, IEEE*, vol. 18, no. 4, pp. 80–92, 2011.

[23] A. Geiger, M. Roser, and R. Urtasun, "Efficient large-scale stereo matching," in *Asian Conference on Computer Vision (ACCV)*, Queenstown, New Zealand, November 2010.

[24] A. Geiger, J. Ziegler, and C. Stiller, "Stereoscan: Dense 3d reconstruction in real-time," in *IEEE Intelligent Vehicles Symposium*, Baden-Baden, Germany, June 2011.