A Framework for the Segmentation and Classification of 3D Point Clouds using Temporal, Spatial and Semantic Information

Mehmet Ali Çağrı Tuncer and Dirk Schulz
Cognitive Mobile Systems, Fraunhofer FKIE, Fraunhoferstr. 20, 53343 Wachtberg, Germany

Keywords: Segmentation, Classification, Support Vector Machines (SVM), Distance Dependent Chinese Restaurant Process, 3D Lidar Data.

Abstract: This paper proposes a novel framework for the segmentation and classification of 3D point cloud which jointly uses spatial, temporal and semantic information. It improves the classification performance by reducing under-segmentation errors. The presented framework, which can determine the number and label of objects in each spatially extracted blob, is decomposed into three steps to acquire spatial, temporal and semantic cues. For the spatial features, blobs are extracted spatially with a neighborhood system on an occupancy grid representation. A smoothed motion field is estimated for the acquisition of temporal cue, where the grid cells are tracked using individual Kalman filters and estimated velocities are transformed to one dimensional movement directions. A support vector machine (SVM) classifier is trained to discriminate the classes of interest for the semantic information of the blobs. A confidence metric is defined to probabilistically compare the volume of each classified blob with the volume of an average object for that class. If this metric is below a predefined threshold, a sequential variant of distance dependent Chinese restaurant process (s-ddCRP) performs the final partition in this blob by using spatial and temporal information. If the s-ddCRP approach splits the blob, the partitioned sub-blobs are afterwards reassigned to new objects by the classifier. Otherwise, the queried blob remains the same. This procedure iteratively continues while searching each blob in the scene at each time frame. Experiments on data obtained with a Velodyne HDL64 scanner in real traffic scenarios illustrate that the proposed framework improves the classification performance of an SVM classifier by reducing under-segmentation errors.

1 INTRODUCTION

Autonomous vehicles require reliable representation and understanding of their environment. The interpretation of sensor readings which provides knowledge of 3D position and movement of dynamic objects in the scene is a fundamental ability for the safe motion of a self-driving car. In order to extract information from 3D Lidar data, perception systems normally go through a point cloud segmentation, object tracking and classification process. The segmentation algorithm is used to cluster different points of the data into smaller blobs according to a similarity criterion. These blobs are subsequently tracked by a tracking algorithm and labeled by a trained classifier into different categories, such as pedestrians, bicycles, cars, etc., over consecutive time frames. Beside the velocity estimation and labeling of blobs, this pipeline predicts the movement of the environment. These predictions are used to plan the autonomous vehicle's own trajectory and to avoid collisions with any obstacles in the surrounding.

The segmentation algorithm of many autonomous vehicles' perception systems relies on simple spatial relationships to group the point cloud into smaller blobs, which represent objects in a scene. A common method is clustering Lidar data together uses their nearness in distance: if points in the data are adequately close to each other, they are assumed to be part of the same object, and if points are far away and disconnected they are assumed to be bound up with different objects.

Well-separated objects can be segregated with an approach using proximity relations alone. However, when individual objects in the point cloud are too close, the segmentation becomes more difficult. For example, pedestrians usually get under-segmented with a neighboring object, such as a parked car or a building. If an autonomous vehicle does not detect that under-segmented pedestrian, the vehicle can

Tuncer, M. and Schulz, D.
A Framework for the Segmentation and Classification of 3D Point Clouds using Temporal, Spatial and Semantic Information.
DOI: 10.5220/0006858903250332
In *Proceedings of the 15th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2018) - Volume 2*, pages 325-332
ISBN: 978-989-758-321-6

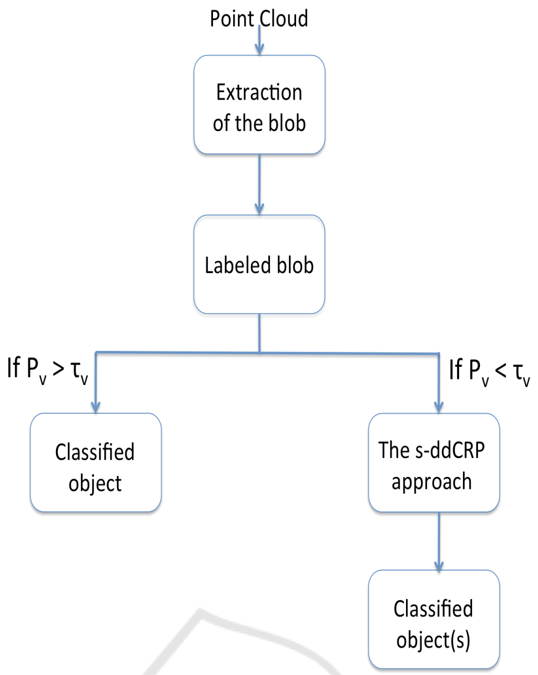

Figure 1: Overview of the proposed framework.

not predict the movements of the pedestrian. Such under-segmentation problems lead to inaccurate or even wrong tracking and classification results, misdetection of objects and, consequently, possible destructive collisions. Therefore, for a more robust object perception process, the segmentation algorithm should benefit from additional cues as well.

This paper proposes a framework for the segmentation and classification of 3D point cloud which jointly uses spatial, temporal and semantic information. It improves the classification performance with overcoming the under-segmentation issue of moving objects, i.e., assigning multiple objects to one blob. When a self-driving vehicle has a complex dynamic environment, such as pedestrians walking close to their nearby objects, detecting if an extracted blob consists of one or multiple objects can be difficult with spatial features alone. This issue leads wrong tracking and classification results. Figure (1) illustrates the overview of the proposed approach. The presented framework, which can determine the number and label of objects in a spatially extracted blob, is decomposed into three steps to acquire spatial, temporal and semantic cues. For the spatial features, the first step is performed on an occupancy grid representation, obtaining connected components of non-ground grid cells, which build up extracted blobs. For the acquisition of temporal cue, a smoothed motion field is estimated for subsequent 3D Lidar scans based on the occupancy grid representation, where the grid cells are tracked using individual Kalman filters and estimated velocities are transformed to one dimensional movement directions. A classification step is applied for the semantic information. Features are extracted from spatially extracted blobs, capturing the distribution of local and global spatial properties. A support vector machine (SVM) classifier (Schölkopf et al., 2000) is trained to discriminate the classes of interest in a supervised learning framework. We defined a confidence metric for the classifier to measure how well a labeled blob matches with its pre-trained class. If the metric is below a threshold, a sequential variant of the distance dependent Chinese Restaurant Process (ddCRP) (Blei and Frazier, 2011) performs the final partition in this blob by using spatial and temporal information. When the s-ddCRP approach partitions the blob, the separated sub-blobs are afterwards reassigned to new objects by the classifier. Otherwise, the assignment and class of the queried blob remains the same. We present experimental results achieved using the data collected with a 3D Velodyne scanner in real traffic to show the feasibility and benefit of the proposed method. Our framework improves the classification performance of an SVM classifier with reducing the under-segmentation errors.

The layout of this paper is as follows. Section 2 discusses the related work. Section 3 explains the extraction of spatial and temporal features. Section 4 presents the proposed segmentation and classification framework in detail. Section 5 evaluates the performance of the presented framework on real traffic data. Section 6 recapitulates the most important findings and gives an outlook on future work.

2 RELATED WORK

Many 3D Lidar based multi-target tracking methods (Petrovskaya and Thrun, 2009; Morton et al., 2011; Teichman et al., 2011; Azim and Aycard, 2012; Choi et al., 2013) use only spatial distance between points for the segmentation of Lidar data, ignoring temporal and semantic information. Therefore these methods may not be able to resolve under-segmentation errors.

Object segmentation and classification have been studied for years (Himmelsbach et al., 2009; Serna and Marcotegui, 2014; Douillard et al., 2014; Habermann et al., 2013). Some methods incorporate semantic evidence for segmentation (Gupta et al., 2014; Lai et al., 2012; Spinello et al., 2010). They can only segment objects of particular classes. Wang et al. (Wang et al., 2012) proposes a classifier for the recognition and segmentation of objects. A probabilistic 3D segmentation method is proposed in (Held et al., 2016) which combines spatial, temporal, and semantic cues to solve under- and over-segmentation problems. Kundu et al. (Kundu et al., 2014) and Sengupta et al. (Sengupta et al., 2013) label each individual point in the scene. A bottom-up approach was proposed by Himmelsbach and Wuensche (Himmelsbach and Wuensche, 2012) that considers the tracking history and appearance of targets for the discrimination of static and dynamic objects. Tuncer and Schulz (Tuncer and Schulz, 2015) proposed the distance dependent Chinese Restaurant Process (ddCRP) (Blei and Frazier, 2011) for the segmentation of 3D Lidar data to exploit spatial and motion features together. The ddCRP is a distribution over partitions of data and based on the Chinese Restaurant Process (CRP) (Pitman et al., 2002). For a faster approach, a sequential variant of ddCRP was proposed, called sequential-ddCRP (s-ddCRP) (Tuncer and Schulz, 2016b). In (Tuncer and Schulz, 2016a), the s-ddCRP segmentation approach is integrated with a smoothed motion field estimation and an object tracking module. In (Tuncer and Schulz, 2017), the mean-shift (Fukunaga and Hostetler, 1975; Comaniciu and Meer, 2002) and ddCRP algorithms were jointly used to significantly decrease the computational time. The presented framework in this paper performs the classification and s-ddCRP segmentation approaches to improve the classification performance of the system with decreasing under-segmentation errors.

3 EXTRACTION OF BLOBS AND TEMPORAL INFORMATION

This section first describes the pre-processing of 3D point cloud measurements, which builds up coarsely extracted blobs. Then we explain the acquisition of temporal information, which consists of grid cell association, Kalman filtering and a smoothing process. The spatially extracted blobs and estimated temporal information are exploited in the segmentation and classification processes by using the proposed framework which is described in Section 4.

3.1 Blob Extraction

We applied the pre-processing approach of (Tuncer and Schulz, 2016a), which briefly consists of occupancy grid representation, filtering and smoothing. The grid cells store the center of mass, averaged heights, variance of the height and total number of the points falling into each grid cell. After the data points belonging to the ground are removed with a decision rule, a connected components algorithm (Bar-Shalom, 1987) using the 8 neighborhood on the grid representation is applied to extract blobs spatially. The framework assigns these coarse blobs to objects and labels them if the confidence metric of the classification method, which will be described in Section 4, is above a threshold. Otherwise, the blobs are sent to the s-ddCRP segmentation algorithm for further partition. Separated sub-blobs are afterwards reassigned and relabeled by the classifier.

3.2 Temporal Information

The 3D point cloud data provides spatial features as described in Subsection 3.1 but the temporal evidence needs to be acquired. Therefore a motion field estimation approach is used to acquire the temporal information. Grid cells are treated as the basic elements of motion and each cell is assigned to its own motion vector. Nearest Neighbor (NN) filters with gating are used for grid cell associations. Individual Kalman filters are applied to each non-ground grid cell to solve the estimation problem. Association errors of grid cells are compensated with a smoothing process as explained in (Tuncer and Schulz, 2016a).

4 THE PROPOSED FRAMEWORK

This section explains our novel framework, which consists of a trained support vector machine (SVM) classifier (Schölkopf et al., 2000) and a variant of sequential distance dependent Chinese Restaurant Process (ddCRP) (Tuncer and Schulz, 2016a), for the segmentation and classification of 3D point clouds by jointly using spatial, temporal and semantic information. Points in the data are clustered into coarse blobs according to the method described in Section 3. An SVM classifier is trained for different categories, such as pedestrians, bicycles, cars, etc. For each trained class, we model the volume of an average object for that class. The extracted blobs are temporarily labeled by the SVM classifier into different categories. A confidence metric is defined to probabilistically compare the volume of each classified blob with the volume of an average object for that class. If this metric is below a pre-defined threshold, a sequential variant of ddCRP performs the final partition in this blob by using spatial and temporal information. If the s-ddCRP approach separates the blob into sub-blobs, they are reassigned to new objects and relabeled by the classifier. Otherwise, the assignment and class of the queried blob remains the same. After the proposed framework has been applied to each blob in a time frame, the algorithm outputs the segmented and classified scene. The cooperation of the classifier and s-ddCRP approach improves the segmentation and classification performances of the object perception system of an autonomous vehicle as shown in Section 5.

4.1 Classification

For the classification of blobs, a set of discriminative features in the data needs to be chosen that either represent the object on point or object level. The extracted features should be representative, ie., similar for objects in a given class, but also discriminative, i.e., vary as much as possible between different classes. The object and point level features are described below.

Object Level Features: Object level features do not involve any local computation of points. The final set contains six features describing global attributes of the blob, all of which are scalar valued.

- f_1: Volume of the blob.
- f_2: Height of the blob.
- f_3: Width of the blob.
- f_4: Length of the blob.
- f_5: Standard deviation of the distance from each point to the center of mass of the blob.
- f_6: Length of the hypotenuse between the width and length of the blob.

Point Level Features: Object level features do not involve local point cloud statistics. Therefore Lalonde features L_1, L_2 and L_3 (Lalonde et al., 2006) are evaluated at all points of the blob. They uses the distribution of neighboring points to express the scatterness, linearness, and surfaceness as explained in (Himmelsbach et al., 2009). The quired point's 20 nearest neighbors within a radius of 0.5m are calculated by constructing a kD-tree.

- L_1: Scaterness.
- L_2: Linearness.
- L_3: Surfaceness.

A histogram for every point feature is defined to be represented in object level. All point features are normalized to take values in the range of 0....1 with dividing every bin value by the total number of points in the blob. Three histograms, each consisting of 4 bins equally spaced over the range 0...1, are added to the final feature set.

We finally have an 18 dimensional feature set, which is defined as $f = (f_1, f_2, f_3, f_4, f_5, f_6, H^4_{L_1}, H^4_{L_2}, H^4_{L_3})$. There are six scalar object level features and three histograms over point level features, each contributing four bins.

For the classification step, a Support Vector Machine (SVM) classifier (Schölkopf et al., 2000) is trained on KITTI data set (Geiger et al., 2012; Fritsch et al., 2013; Geiger et al., 2013) which provides labeled objects from different time steps. For the multiclass problem, a *one-versus-all* approach is applied, where one binary SVM is trained for every class, separating the class from all other classes. The SVM depends on a penalty parameter ζ for weighting classification errors and a kernel function parameter γ. A grid search is performed to determine the optimal choice of these parameters. In the grid search, different pairings of ζ and γ are validated and the parameters with the best performance are chosen.

We apply the validation set method, which simply divides the labeled data in a training set which is used to determine parameters and a validation set that evaluates the performance.

After the SVM classifier is trained for different categories, the volumetric size of an average object for each class is modeled by a Gaussian with parameters μ_c and σ_c. A confidence metric is defined in Equation (1) to probabilistically compare the volumetric size of each classified blob, b_v, with the volume of an average object o_b for that class.

$$P_V(b_v|o_b) = \eta \exp\left(\frac{-(b_v - \mu_{o_b})}{2\sigma^2_{o_b}}\right) < \tau_v \qquad (1)$$

If the metric in Equation (1) is below the threshold τ_v for a labeled blob, a sequential variant of ddCRP defined in Subsection 4.2 decides the final clustering in this blob by incorporating spatial and temporal features. The divided sub-blobs are afterwards reassigned to new objects and relabeled by the classifier. Otherwise, the assignment and class of the queried blob remains the same.

4.2 Sequential Distance Dependent Chinese Restaurant Process

After mapping the data on a grid and removing the points belonging to the ground, we apply a connected components algorithm on the occupancy grid to spatially partition the scene into blobs. We applied the sequential distance dependent Chinese restaurant process (s-ddCRP), which was proposed in (Tuncer and Schulz, 2016b) to partition the blobs by grouping the grid cells together with considering spatial continuity and the features of grid cells. The s-ddCRP approach determines the clusters, which represent different objects, in a blob based on posterior inference. Different segmentation hypotheses are generated and the s-ddCRP decides on the most probable ones by using temporal and spatial features together. The mean value of smoothed motion vectors of grid cells which form an object can be assigned as a temporal feature of the object. The scope of this paper is on the integration of segmentation and classification steps.

5 EXPERIMENTAL RESULTS

The proposed framework was evaluated on the KITTI data set (Geiger et al., 2012; Fritsch et al., 2013; Geiger et al., 2013) which was recorded using a Velodyne HDL-64D Lidar sensor from a moving car on city streets. It consists of tracklets, which are the sequences of the same objects from different time steps in a recording. We used approximately 80% of these tracklets to train our method and select parameters, and the remaining tracklets were used for testing and evaluation. Table (1) shows the number of examples used in each data set. Tracklets from different data sets were used for training and test to avoid the bias of an object reoccurring in both data sets.

The SVM depends on a penalty parameter ζ for weighting classification errors and a kernel function parameter γ. A grid search was performed to determine the optimal choice of these parameters. In the grid search, different pairings of ζ and γ were validated with $\zeta = 2^{-7}, 2^{-5}, ...2^{9}$ and $\gamma = 2^{-17}, 2^{-15}, ...2^{5}$. The parameters were chosen as $\zeta = 2^{7}$ and $\gamma = 2^{-3}$. Estimated grid cell velocities were transformed to one-dimensional movement directions. For the s-ddCRP part of the proposed framework, larger α values bias the algorithm towards more clusters so we set $\alpha = 10^{-4}$ (Tuncer and Schulz, 2016b). The ddCRP sampler was run with 20 iterations for each extracted blob.

Table 1: Number of samples for each class and for each data set.

Class	Training	Evaluation
Pedestrian	1208	293
Cyclist	556	141
Car	7771	1942
Van	1039	196
Total	10574	2639

Figure (2) shows how the proposed framework runs for the segmentation and classification of 3D Lidar data. It displays a scene where a person goes out of the car and starts walking. The camera image is given for better understanding of the scene. The blob is spatially extracted in the second image. This leads to the under-segmentation of the person with the car. Therefore the SVM classifier labels the blob as a car. When the confidence metric defined in Equation (1) is below the threshold τ_v, the s-ddCRP approach makes a further clustering in this blob by jointly incorporating spatial and temporal features. The divided two sub-blobs are afterwards reassigned to a car and a pedestrian by the classifier. This procedure iteratively continues while searching each blob in the scene at each time frame. After the proposed framework has been applied to each blob in a time frame, the algorithm outputs the labeled scene.

Table 2: The number of under-segmented objects for the classes in the test data set.

	Pedestrian	Cyclist	Car
Spatial Alone	60	31	15
Proposed Framework	13	20	9

Output of the proposed framework is a partitioning of the grid cells in each time step into disjoint labeled blobs, where each blob is intended to represent a single object instance. The evaluation metric defined in (Tuncer and Schulz, 2017) is used to evaluate how well the proposed framework avoids under-segmentation errors. Table (2) shows the number of under-segmented objects for the pedestrian, cyclist and car categories in the test data set. The first row is the number of under-segmented blobs when they are extracted spatially while the second row is the number of under-segmented objects at the output of the proposed framework. It can cope with dynamic under-

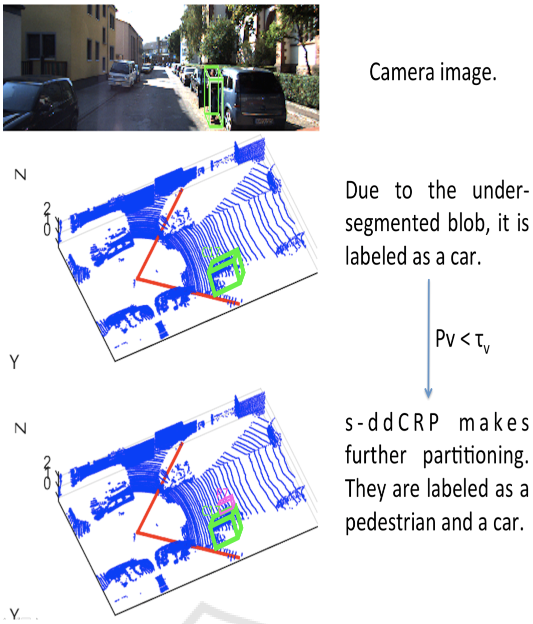
Camera image.
N
Y
Due to the under-segmented blob, it is labeled as a car.
Pv < τ_v
N
Y
s-ddCRP makes further partitioning. They are labeled as a pedestrian and a car.

Figure 2: Levels of the proposed framework for the under-segmented objects.

segmented objects. However, due to the stationary under-segmented objects, which do not have temporal cues, and the group of nearby pedestrians moving in the same direction, the proposed framework can not completely avoid under-segmentation errors.

Table 3: Confusion matrix for the SVM classifier when the spatially extracted blobs are labeled directly without the proposed framework.

	Pedestrian	Cyclist	Car	Van
Pedestrian	225	13	41	14
Cyclist	13	105	7	16
Car	11	7	1915	9
Van	0	3	64	129

Table (3) gives the accuracy results of the SVM classifier in case the extracted blobs are labeled directly without the s-ddCRP approach. The rows are the ground truth labels while the columns show the predicted class labels. When the segmentation and classification processes are performed consecutively, pedestrians and cyclists moving close to other vehicles are mostly under-segmented and labeled as their neighboring objects. The overall accuracy of the classifier stays around 89.9% because of those pedestrians and cyclists which are under-segmented with cars and vans. The other reason of this accuracy rate is the issue of discerning vans from cars because of their similar appearances. More sophisticated features might solve this issue.

The confusion matrix for the classification result of the proposed framework is given in Table (4) to evaluate the benefit of incorporating spatial, temporal and semantic information. The labeled blobs which have low values of confidence metric in the first level of the proposed framework are sent to the s-ddCRP algorithm for further investigation to avoid under-segmentation errors. The s-ddCRP approach does the further segmentation of dynamic pedestrian and cyclist blobs. Afterwards these divided sub-blobs are relabeled into correct classes. Avoiding the under-segmentation errors raises the overall classification accuracy of the proposed framework to 92.5%.

Table 4: Confusion matrix for the classification result of the proposed framework. The rows are the ground truth labels while the columns show the predicted class labels.

	Pedestrian	Cyclist	Car	Van
Pedestrian	272	11	6	4
Cyclist	12	116	2	11
Car	11	7	1921	3
Van	0	3	64	129

The confidence metric is defined in Equation (1) to probabilistically compare the volumetric size of each classified blob with the volume of an average object for that class. If the metric is below the threshold τ_v for a labeled blob, the s-ddCRP makes the final clustering in this blob. Figure (3) illustrates the percentage number of blobs sent to the s-ddCRP algorithm in the proposed framework depending on the threshold value τ_v of the confidence metric. Considering the total number of under-segmented objects, which is 4% of the whole segments in the test set, the volumetric model defined in Equation (1) causes unnecessary furt-

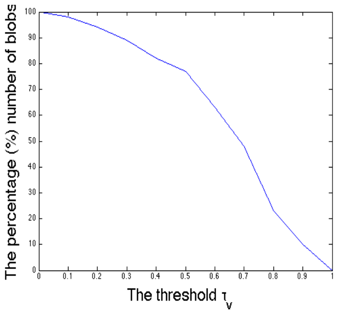

Figure 3: The percentage number of blobs sent to the s-ddCRP algorithm in the proposed framework depending on the threshold value τ_v of the confidence metric.

her investigations of blobs, which does not change the blob structures. A more complex confidence metric might prevent this issue.

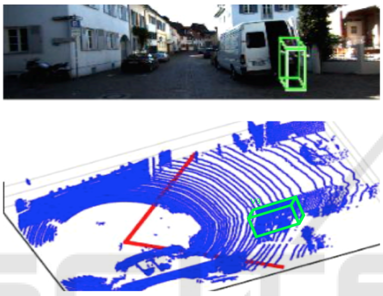

Figure 4: Under-segmented pedestrian and van due to the lack of temporal information.

Figure (4) shows that there is a pedestrian standing behind a stationary van. This person is correctly detected with a bounding box in the camera image. However, the person has no temporal information so our framework detects them as one object in the bottom image. Because of stationary under-segmented objects, which do not have temporal cues, and the group of nearby pedestrians moving in the same direction, the framework can not cope with all under-segmentation errors as shown in Table (2). Beside the usage of spatial and temporal information, incorporating semantic clues would improve the segmentation results, and, thus, will be part of our future work.

6 CONCLUSION

We proposed a framework for the segmentation and classification of 3D point cloud which jointly uses spatial, temporal and semantic information to improve classification performance with overcoming the under-segmentation errors. Reduction of the motion estimation into one dimension is sufficient to discriminate moving objects from their neighbors such as parked cars. However, distinguishing stationary under-segmented objects and the group of pedestrians moving in the same direction still remains as a problem. An appearance model together with the spatial and temporal features might help to solve this issue.

Our framework uses spatial and semantic cues for the classification. Afterwards it exploits semantic features to decide if the blob needs further segmentation. However, the further segmentation is done with spatial and temporal information. Adding semantic cues for the segmentation process would significantly resolve the under-segmentation problem of stationary nearby objects and, thus, improve the general performance of the object perception system.

The confidence metric is defined in Equation (1) to probabilistically compare the volumetric size of each classified blob with the volume of an average object for that class. If the metric is below a threshold for a labeled blob, the s-ddCRP makes the final clustering in this blob. We check the confidence value of the classifier one time for each blob in the proposed framework. It is illustrated that the volumetric model causes unnecessary further investigations for the segmentation of blobs, which does not change the blob structures and increase the computational time. A more sophisticated confidence metric might prevent this problem.

This paper mostly focuses on improving classification performance with providing better segmentation of the scene. However, we noticed that the classifier discerns vans from cars with high error rates due to their similar appearances. Therefore more complementary descriptive features should be proposed to overcome this problem.

Applying an iterative closest point approach would be interesting instead of tracking each grid cell on an occupancy grid. We intend to compare the performance of our framework with other novel algorithms proposed in the literature.

REFERENCES

Azim, A. and Aycard, O. (2012). Detection, classification and tracking of moving objects in a 3d environment. In *Intelligent Vehicles Symposium (IV), 2012 IEEE*, pages 802–807. IEEE.

Bar-Shalom, Y. (1987). *Tracking and data association*. Academic Press Professional, Inc.

Blei, D. M. and Frazier, P. I. (2011). Distance dependent chinese restaurant processes. *The Journal of Machine Learning Research*, 12:2461–2488.

Choi, J., Ulbrich, S., Lichte, B., and Maurer, M. (2013). Multi-target tracking using a 3d-lidar sensor for autonomous vehicles. In *Intelligent Transportation Systems-(ITSC), 2013 16th International IEEE Conference on*, pages 881–886. IEEE.

Comaniciu, D. and Meer, P. (2002). Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on pattern analysis and machine intelligence*, 24(5):603–619.

Douillard, B., Underwood, J., Vlaskine, V., Quadros, A., and Singh, S. (2014). A pipeline for the segmentation and classification of 3d point clouds. In *Experimental robotics*, pages 585–600. Springer.

Fritsch, J., Kuhnl, T., and Geiger, A. (2013). A new performance measure and evaluation benchmark for road detection algorithms. In *Intelligent Transportation Systems-(ITSC), 2013 16th International IEEE Conference on*, pages 1693–1700. IEEE.

Fukunaga, K. and Hostetler, L. (1975). The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Transactions on information theory*, 21(1):32–40.

Geiger, A., Lenz, P., Stiller, C., and Urtasun, R. (2013). Vision meets robotics: The kitti dataset. *International Journal of Robotics Research (IJRR)*.

Geiger, A., Lenz, P., and Urtasun, R. (2012). Are we ready for autonomous driving? the kitti vision benchmark suite. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3354–3361. IEEE.

Gupta, S., Girshick, R., Arbeláez, P., and Malik, J. (2014). Learning rich features from rgb-d images for object detection and segmentation. In *European Conference on Computer Vision*, pages 345–360. Springer.

Habermann, D., Hata, A., Wolf, D., and Osório, F. S. (2013). Artificial neural nets object recognition for 3d point clouds. In *Intelligent Systems (BRACIS), 2013 Brazilian Conference on*, pages 101–106. IEEE.

Held, D., Guillory, D., Rebsamen, B., Thrun, S., and Savarese, S. (2016). A probabilistic framework for real-time 3d segmentation using spatial, temporal, and semantic cues. In *Proceedings of Robotics: Science and Systems*.

Himmelsbach, M., Luettel, T., and Wuensche, H.-J. (2009). Real-time object classification in 3d point clouds using point feature histograms. In *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, pages 994–1000. IEEE.

Himmelsbach, M. and Wuensche, H.-J. (2012). Tracking and classification of arbitrary objects with bottom-up/top-down detection. In *Intelligent Vehicles Symposium (IV), 2012 IEEE*, pages 577–582. IEEE.

Kundu, A., Li, Y., Dellaert, F., Li, F., and Rehg, J. M. (2014). Joint semantic segmentation and 3d reconstruction from monocular video. In *European Conference on Computer Vision*, pages 703–718. Springer.

Lai, K., Bo, L., Ren, X., and Fox, D. (2012). Detection-based object labeling in 3d scenes. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 1330–1337. IEEE.

Lalonde, J.-F., Vandapel, N., Huber, D. F., and Hebert, M. (2006). Natural terrain classification using three-dimensional ladar data for ground robot mobility. *Journal of field robotics*, 23(10):839–861.

Morton, P., Douillard, B., and Underwood, J. (2011). An evaluation of dynamic object tracking with 3d lidar. In *Proc. of the Australasian Conference on Robotics & Automation (ACRA)*.

Petrovskaya, A. and Thrun, S. (2009). Model based vehicle detection and tracking for autonomous urban driving. *Autonomous Robots*, 26(2-3):123–139.

Pitman, J. et al. (2002). Combinatorial stochastic processes. *Technical Report 621, Dept. Statistics, UC Berkeley, 2002. Lecture notes for St. Flour course*.

Schölkopf, B., Smola, A. J., Williamson, R. C., and Bartlett, P. L. (2000). New support vector algorithms. *Neural computation*, 12(5):1207–1245.

Sengupta, S., Greveson, E., Shahrokni, A., and Torr, P. H. (2013). Urban 3d semantic modelling using stereo vision. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 580–585. IEEE.

Serna, A. and Marcotegui, B. (2014). Detection, segmentation and classification of 3d urban objects using mathematical morphology and supervised learning. *ISPRS Journal of Photogrammetry and Remote Sensing*, 93:243–255.

Spinello, L., Arras, K. O., Triebel, R., and Siegwart, R. (2010). A layered approach to people detection in 3d range data. In *AAAI*, volume 10, pages 1–1.

Teichman, A., Levinson, J., and Thrun, S. (2011). Towards 3d object recognition via classification of arbitrary object tracks. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 4034–4041. IEEE.

Tuncer, M. A. Ç. and Schulz, D. (2015). Monte carlo based distance dependent chinese restaurant process for segmentation of 3d lidar data using motion and spatial features. In *Information Fusion (FUSION), 2015 18th International Conference on*, pages 112–118. IEEE.

Tuncer, M. A. Ç. and Schulz, D. (2016a). Integrated object segmentation and tracking for 3d lidar data. In *Proceedings of the 13th International Conference on Informatics in Control, Automation and Robotics - Volume 2: ICINCO*, pages 344–351.

Tuncer, M. A. Ç. and Schulz, D. (2016b). Sequential distance dependent chinese restaurant processes for motion segmentation of 3d lidar data. In *Information Fusion (FUSION), 2016 19th International Conference on*, pages 758–765. IEEE.

Tuncer, M. A. . and Schulz, D. (2017). A hybrid method using temporal and spatial information for 3d lidar data segmentation. In *Proceedings of the 14th International Conference on Informatics in Control, Automation and Robotics - Volume 2: ICINCO*, pages 162–17.

Wang, D. Z., Posner, I., and Newman, P. (2012). What could move? finding cars, pedestrians and bicyclists in 3d laser data. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 4038–4044. IEEE.