18th International Conference on Information Fusion
Washington, DC - July 6-9, 2015

Monte Carlo Based Distance Dependent Chinese Restaurant Process for Segmentation of 3D LIDAR Data Using Motion and Spatial Features

Mehmet Ali Cagri Tuncer
Cognitive Mobile Systems
Fraunhofer FKIE
Wachtberg, Germany 53343
Email: mehmet.tuncer@fkie.fraunhofer.de

Dirk Schulz
Cognitive Mobile Systems
Fraunhofer FKIE
Wachtberg, Germany 53343
Email: dirk.schulz@fkie.fraunhofer.de

***Abstract*—This paper proposes a novel method to obtain robust and accurate object segmentations from 3D Light Detection and Ranging (LIDAR) data points. The method exploits motion information simultaneously estimated by a tracking algorithm in order to resolve ambiguities in complex dynamic scenes. Typical approaches for tracking multiple objects in LIDAR data follow three steps; point cloud segmentation, object tracking, and track classification. A large number of errors is due to failures in the segmentation component, mainly because segmentation and tracking are performed consecutively and the segmentation step solely relies on geometrical features. This article presents a 3D LIDAR based object segmentation method that exploits the motion information provided by a tracking algorithm and spatial features in order to discriminate spatially close objects. After a pre-processing step that maps LIDAR measurements to an occupancy grid representation, the motions of grid cells are estimated using independent Kalman filters. A distance dependent Chinese Restaurant Process based Markov chain Monte Carlo approach is applied to generate different segmentation hypotheses and decide on the most probable segments by using motion and spatial features together.**

I. Introduction

Autonomous vehicles are capable of sensing their dynamic and static environments, which is very crucial for collision avoidance. Recent sensors like 3D Light Detection and Ranging (LIDAR) [1] provide large amounts of point cloud data for perception of arbitrary objects in the vehicle's surroundings. These sensors directly provide 3D position information of the surface of objects and in contrast to cameras their performance is independent of the ambient light conditions; so night time operation, cloudy weather or shadowy areas do not affect their operation. These advantages make LIDAR a suitable sensor for environment perception.

However, in order to perceive the dynamics in the vehicle's environment, objects need to be tracked in the high volume point cloud data. Typically, Multi Target Tracking (MTT) methods [2]–[5] are applied to estimate the states of all detected targets such as vehicles, pedestrians, bicyclists. These methods follow a three step approach encompassing point cloud segmentation, object tracking and track classification. MTT methods perform object segmentation step first, which tends to unavoidable segmentation errors. These errors in turn cause erroneous track estimates. So improving the segmentation process is important to achieve progress on the whole recognition and tracking process.

In urban scenarios, traffic participants are assumed well segmented from each other. Therefore, current point cloud segmentation approaches rely on geometrical features only. However the assumption may not hold, especially for bicyclists and pedestrians. They often get close to other objects such as parking cars. Under these circumstances, segmentation gets difficult and under-segmentation of objects can occur.

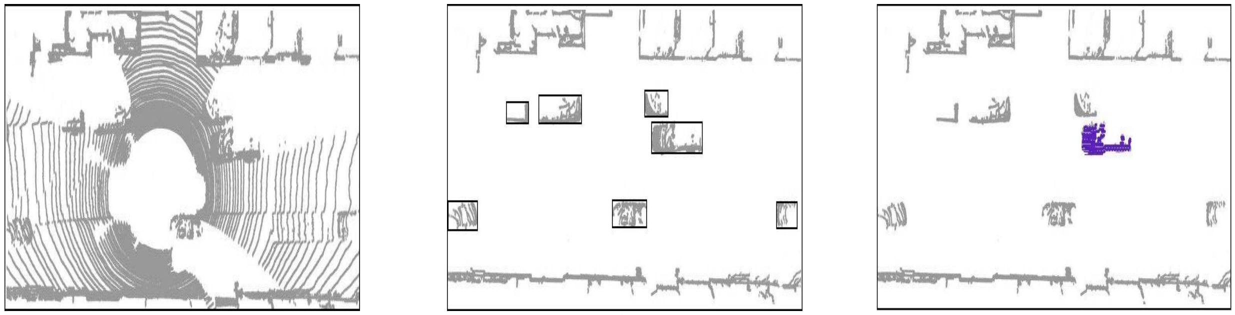
In this paper, we present a novel method to obtain robust and accurate object segmentations from 3D LIDAR data points. The motivation of our approach is exploiting the motion and spatial information simultaneously to discriminate close objects. In order to obtain motion features, we first address how a motion field of the surrounding environment is formed using 3D LIDAR measurements. Then a distance dependent Chinese Restaurant Process (ddCRP) [6] is adapted in order to use both spatial and motion features together for the segmentation of the point clouds. The Markov chain Monte Carlo based ddCRP method models spatial dependencies and enforces spatial contiguity of the inferred parts.

This paper is organized as follows. Section II starts with a discussion of previous related work. Section III explains the pre-processing of raw measurements and estimation of motion features. Section IV shows how the proposed Bayesian non-parametric method is used for segmentation of close objects. In Section V the performance of the proposed framework is evaluated with real data. Section VI concludes the paper and gives an outlook on future work.

II. Related Work

LIDAR based MTT approaches [2]–[5] perform the segmentation and tracking components consecutively and use only spatial features in order to segment scenes. Himmelsbach and Wuensche [7] apply a bottom-up approach by considering the appearance and tracking history of targets to discern static and moving objects.

978-0-9964527-1-7©2015 ISIF



(a) (b) (c)

Fig. 1. (a) Raw data, (b) Non-ground measurements. Segmentation blobs in black boxes represent potential moving objects (c) Purple regions show the estimated motion field of the scene

In order to estimate motion states in the scene, we address the Visual Motion Field Estimation (VMFE) methods [8]–[10]. They are applied to determine motion vectors in a scene using colour information. VMFE methods are capable of tracking all dynamic scenes without prior segmentation. This avoids errors of perception systems due to wrong segmentations. Recently several 3D VMFE methods were proposed [11]–[13]. However visual sensors are not able to provide 3D positions directly, so obtaining 3D positions from 2D images causes noisy estimates. The advantage of 3D LIDAR over visual systems is that they directly provide 3D positions of points, but they do not provide colour information. In order to exploit MFE method with 3D point cloud data, measurements are mapped to an occupancy grid representation with the assumption of each grid cell having its own motion state. These motion states are estimated using independent Kalman filters. This approach is similar to [14], who also regard the scene as a motion field, but use a spatial smoothing algorithm to get tracking results.

After estimating the motion state of each grid cell and doing segmentation according to their spatial correlations, the problem turns into determining the number of subsegment regions representing different objects in a single segmentation blob. We expect to find spatially contiguous grid cells by using their motion and spatial features. A distance dependent Chinese Restaurant Process (ddCRP) [6] was adapted to solve this clustering problem. The ddCRP is an extension of the Chinese Restaurant Process (CRP) [15]. The CRP relies on a exchangeability assumption, which is necessary because this model is based on a Dirichlet process. The CRP and its connection to the Dirichlet process is described in [16]. The ddCRPs make no exchangeability assumption and are therefore capable of modelling spatial relations. For instance, nearby grid cells are more likely to cluster together. In [17], the ddCRP is used as a nonparametric clustering technique and combined with a spectral dimensionality reduction method. Ghosh *et al.* [18] apply the spatial distance dependent CRP for natural image segmentation. In [19], potentially unbounded parts of an articulated object from aligned meshes in different poses are discovered with the ddCRP approach.

III. Motion State Estimation

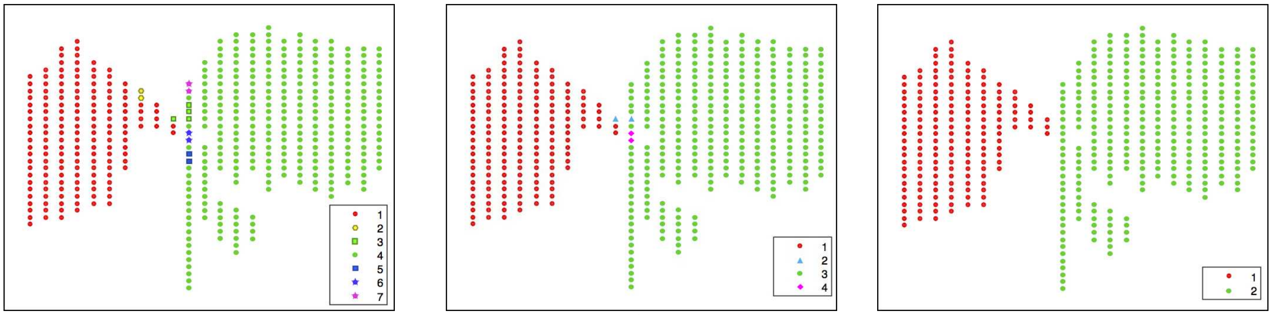
Recent 3D sensors provide large amounts of point cloud data for perception of arbitrary dynamic objects. The data used in this paper is gathered by a Velodyne HDL-64D LIDAR [1]. This sensor has a frame rate of 10 Hz and a 360 degree horizontal field of view. It produces approximately 1.1 million point measurements per second, which makes it impossible to exploit MFE methods directly. Therefore, measurements are mapped to an occupancy grid $\mathbf{Gr}$ to reduce the amount of data. All points at time t are projected to grid cells $\mathbf{gr}_t = \{gr_{t,1}, gr_{t,2}, ...gr_{t,N}\}$ where $gr_{t,i} = (gr_x, gr_y)$ represents the center of mass of points in the grid cell. The average height of points in a grid cell is used to remove points belonging to the ground. Grid cells are assumed to be the basic elements for motion estimation, which means that each grid cell is assumed to have its own velocity vector. After a connected components algorithm using an 8 neighborhood on the grid is invoked to extract segmentation blobs, thresholding on the size of the extracted blobs is applied to prune huge blobs. This avoids tracking the grid cells of big static structures such as buildings. The result of these steps is illustrated in Figure (1) (a) and Figure (1) (b). The extracted segmentation blobs are checked for a necessary sub-segmentation using the ddCRP introduced later in Section IV. First, we explain the motion state vector $\mathbf{x}_t$ estimation, which consists of a data association and a velocity vector estimation part.

A. Data Association

This section explains the process of associating grid cells of the previous and current scans. A Nearest Neighbor (NN) filter is used for the data association process. Predicting the cell locations from state vectors of the previous scan yields a validation gate. If there are measurements lying in the gate, the closest neighbor measurement is accepted based on Euclidian distances. Otherwise the current grid cell is not associated with any predicted measurements and a new Kalman filter is generated for that cell.

B. Kalman Filter

In order to solve the state vector estimation problem, independent linear Kalman filters are applied to each grid cell



(a) (b) (c)

Fig. 2. Effect of the scaling parameter α on the cluster structure of a segmentation blob which contains two close objects. Points represent the grid cells (a) when $\alpha = 10$, (b) when $\alpha = 1$, (c) when $\alpha = 10^{-4}$

in the current scan. If a grid cell $gr_{t,i}$ is associated with $gr_{t-1,i}$ from the previous scan, the predicted state of $gr_{t-1,i}$ is used to initialize the motion state of the grid cell $gr_{t,i}$ in the current scan, which is then updated with the coordinates of $gr_{t,i}$. The state vector $\mathbf{x}_t$ at time t is evolved from the state vector $\mathbf{x}_{t-1}$ according to the Kalman filter model in Equation (1).

$$\mathbf{x}_t \;=\; \mathbf{F}_t \mathbf{x}_{t-1} \;+\; \mathbf{w}_t \tag{1}$$

$\mathbf{F}_t$ is the time invariant state transition model and $\mathbf{w}_t$ is the process noise, $\mathbf{w}_t \backsim N\left(0, \mathbf{Q}_t\right)$ with covariance $\mathbf{Q}_t$. The state vector $\mathbf{x}_t^T$ contains the estimated center of mass of the points in the grid cell and their corresponding velocities $\left[\hat{gr}_x, \hat{gr}_y, v_x, v_y\right]$. An observation $\mathbf{gr}_t$ of the state vector $\mathbf{x}_t$ at time t is explained by

$$\mathbf{gr}_t \;=\; \mathbf{H}_t \mathbf{x}_t \;+\; \mathbf{v}_t \tag{2}$$

where $\mathbf{H}_t$ is the time invariant observation model and $\mathbf{v}_t$ is the observation noise, $\mathbf{v}_t \backsim N\left(0, \mathbf{R}_t\right)$ with covariance $\mathbf{R}_t$.

IV. Segmentation with Distance Dependent CRP

In Section III nearby grid cells are extracted into segmentation blobs based on the Euclidean distance only. If objects get close to each other, under-segmentation problems can occur. The motivation of our approach is to exploite motion and spatial features simultaneously in order to discriminate the close objects. For this purpose we employ a non-parametric Bayesian clustering technique. Adjacent grid cells in a segmentation blob with similar state vectors $\mathbf{x}_t$ are grouped to subsegments, or clusters $z\left(c_i\right)$, which then form the individual objects in a blob. To extract these objects, we need to find contiguous regions of grid cells assigned to the same cluster and simultaneously determine the unknown number of clusters in the extracted blobs. A distance dependent Chinese Restaurant Process (ddCRP) is adapted to solve this clustering problem.

The ddCRP is an extension of the Chinese Restaurant Process (CRP). The CRP is typically introduced as a customer to table assignment process in a restaurant with a potentially infinite number of tables. Costumers enter the restaurant one by one and costumer i sits down at a table $z\left(c_i\right)$ with a probability proportional to the number of people already sitting at that table or picks up a new table with a probability proportional to a scaling parameter α. At the end of the process, the occupied tables yield the partition of the data.

The CRP is an exchangeable model, i.e. the order of the observed data does not affect the posterior distribution over partitions. However, exchangeability does not hold for the segmentation of extracted blobs, because the coordinates of grid cells need to be considered to obtain contiguous regions.

Unlike the CRP, the distance dependent CRP provides a method to model features and non-exchangeability by linking customers to other customers instead of tables. Customers i and j sit together with a probability proportional to a decreasing function of the distances $f(d_{ij})$ or customer i can sit alone with a probability proportional to α. This is described in Equation (3).

$$p\left(c_i = j | D, f, \alpha\right) \propto \begin{cases} f(d_{ij}) & \text{if } i \neq j, \\ \alpha & \text{if } i = j. \end{cases} \tag{3}$$

In our case the grid cells are the customers and a window decay function $f(d) = 1\left[d < a\right]$ is used to connect grid cells. This function enforces the algorithm to constitute spatially connected clusters in a segmentation blob. An example of the effect of the parameter α on the cluster structure of a segmentation blob is given in Figure (2). A smaller α pushes the algorithm towards larger clusters.

A mixture model of a restaurant can be defined with a base distribution G_0, which is a distribution over cluster distributions. Then the data are drawn as follow:

1) For each customer, sample an associated link $c_i \backsim ddCRP(\alpha, f, D)$. Tables $z\left(\mathbf{c}\right)$ are deterministic assignments of the sampled links $\mathbf{c} = \left[c_1, c_2, .., c_N\right]$
2) For each table, draw parameters $\boldsymbol{\theta}_k \backsim G_0$
3) For each customer, sample data $\mathbf{x}_i \backsim F(\boldsymbol{\theta}_k)$

Note that in this section we leave out the time indices t of $\mathbf{x}$ for simplicity so $\mathbf{x}_i$ denotes the state vector of grid i in the current time frame t.

In our problem, the restaurant represents each extracted segmentation blob from the scene, tables denote the clusters,

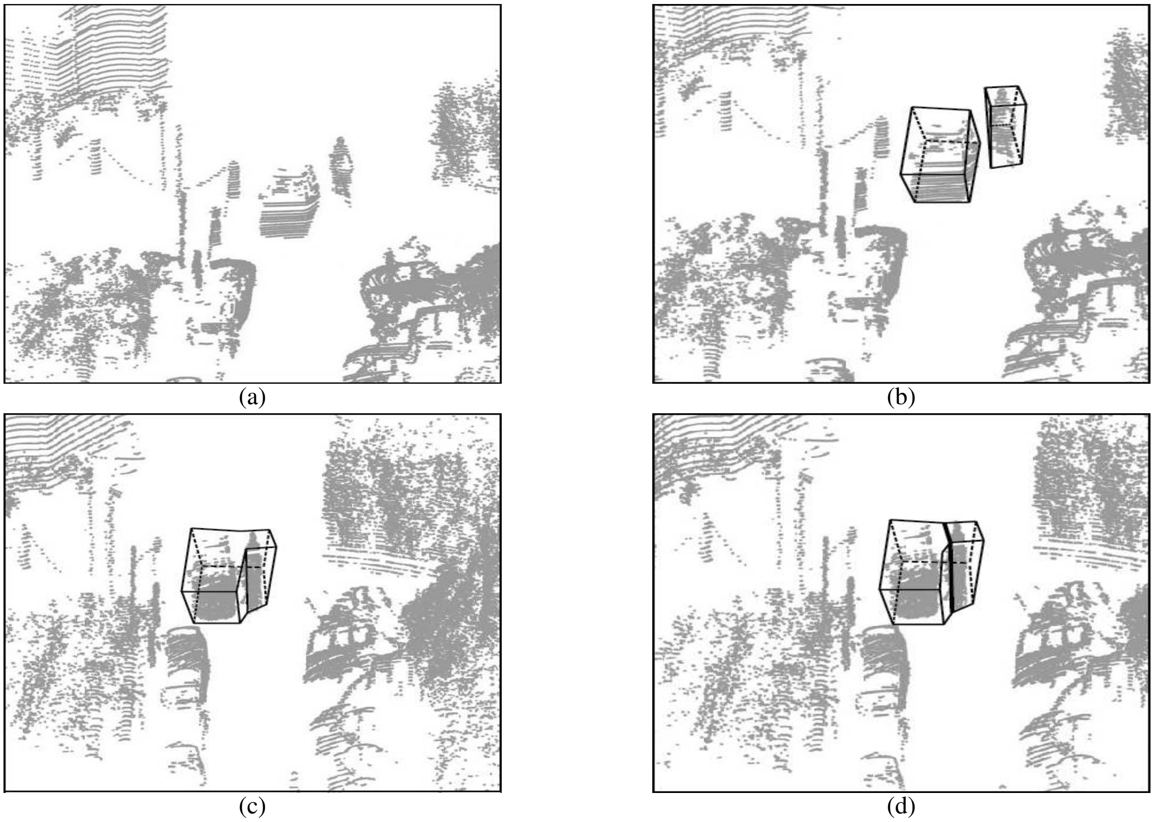


Fig. 3. (a) A bicyclist is getting closer to a parking car (b) For visualization, a bounding box is fitted to the objects which are considered. (c) The bicyclist and the car are extracted in a segmentation blob as one object by using spatial features only (d) The bicyclist and the car are discriminated successfully by the proposed ddCRP based segmentation method. A thick solid line shows the discrimination of the segmentation blob into two partitions which belong to the bicyclist and the parking car.

or objects in a segmentation blob, and customers are the grid cells.

Adjacent grid cells are linked together using Equation (3). Grid cell assignments are used to sample estimated motion features. The posterior distribution of grid cell assignments conduces to a posterior over clusters, which provides the number of objects in a segmentation blob.

A. Posterior Inference

Objects in segmentation blobs can be found by posterior inference which determines the conditional distribution of the hidden variables given the observations. However the posterior is intractable due to huge combinatorial number of possible customer layouts [6]. Therefore Gibbs sampling [20], a Markov Chain Monte Carlo (MCMC) sampling [21], is used for the inference. We iteratively sample each latent variable c_i given other latent variables $\mathbf{c}_{-i}$ and observations $\mathbf{x}$ as in Equation (4).

$$p\left(c_i|\mathbf{c}_{-i},\mathbf{x},\Omega\right) \propto p\left(c_i|D,\alpha\right) p\left(\mathbf{x}|z\left(\mathbf{c}\right),\mathbf{\Theta}\right) \tag{4}$$

where $\Omega = \{D, \alpha, \mathbf{\Theta}\}$ is the hyperparameters. D is distance, α is the scaling factor, and $\mathbf{\Theta}$ is the base distribution parameter. The first term of Equation (4) is given in Equation (3) as the ddCRP prior. The second one is the likelihood term of the observation under the partition $z\left(\mathbf{c}\right)$ where $\mathbf{c} = \left(\mathbf{c}_{-i} \cup c_i\right)$.

Gibbs sampling in Equation (4) is implemented in two stages. First the current link c_i is removed from the cluster configuration. Removing a customer link either splits a cluster or does not affect the current structure. On the one hand, if c_i is the only connection between data point i and its cluster, it splits. On the other hand, if there are alternative connections to the cluster or if c_i is a self-link, clusters stay unchanged.

In the second stage of Gibbs sampling, it is considered how each alternative new link affects the likelihood of the observations by replacing the current customer link c_i. Replacing the customer link either joins the clusters or leaves them unchanged. On the one hand, when reassigning the customer link c_i connects the customers of its cluster with customers from a different cluster, these two clusters are joined. On the other hand, if c_i is a self link or if it is linked to a customer that is already in this cluster under $z\left(\mathbf{c}_{-i}\right)$, the cluster structure remains the same.

The sampler finds out the number of clusters in a segmentation blob by removing and randomly reassigning the customer links, or grid cell assignments in our case. The likelihood term in Equation (5) can be decomposed as shown in Equation (5).

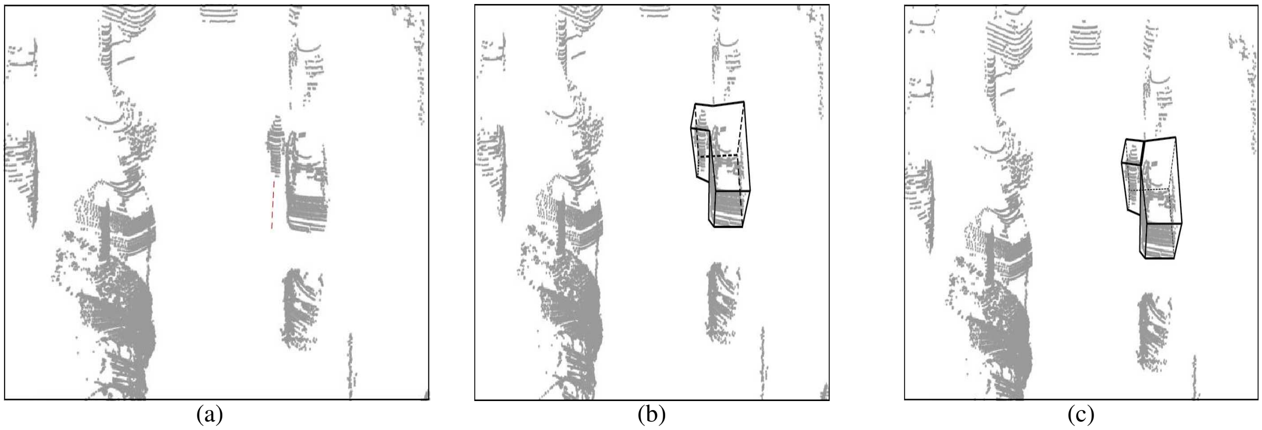


Fig. 4. (a) A person is walking along a parking car. Red dashed lines show the trajectory of the person. (b) A segmentation algorithm which considers only spatial features extracted two objects in the same segmentation blob (c) The proposed ddCRP based segmentation method discerns the person in the blob. A thick solid line shows the separation of the segmentation blob into two partitions which belong to the person and the parking car

$$p\left(\mathbf{x}|z\left(\mathbf{c}\right),\mathbf{\Theta}\right)=\prod_{k=1}^{K}p\left(\mathbf{x}_{z(\mathbf{c})=k}|\mathbf{\Theta}\right)\tag{5}$$

where K is the number of clusters, and $\mathbf{x}_{z(\mathbf{c})=k}$ denotes the observations assigned to cluster k. Observations at each cluster are sampled independently by using the parameters drawn from the base distribution G_0. Then the marginal probability is computed as follows,

$$p\left(\mathbf{x}_{z(\mathbf{c})=k}|\mathbf{\Theta}\right)=\int\left(\prod_{i\in z(\mathbf{c})=k}p\left(x_i|\boldsymbol{\theta}\right)\right)p\left(\boldsymbol{\theta}|\mathbf{\Theta}\right)d\mathbf{\Theta}\tag{6}$$

Here i denotes the indices assigned to cluster k and $\mathbf{\Theta}$ is the parameter of the base distribution G_0. If $p\left(x_i|\boldsymbol{\theta}\right)$ and G_0 are selected as conjugate, this enables the marginalization of the cluster parameter $\boldsymbol{\theta}$ in order to compute Equation (6) analytically [22].

In Gibbs sampling, we need to compute cases that alter the cluster structure. Considering that m and l denote the cluster indices joined to cluster k, we can specify a Markov chain as follows,

$$p\left(c_i|\mathbf{c}_{-i},\mathbf{x},\Omega\right)\propto\begin{cases}\Lambda\left(\mathbf{x},z,\mathbf{\Theta}\right) & \text{if } c_i \text{ joins } m \text{ and } l,\\ p\left(c_i|D,\alpha\right) & \text{otherwise.}\end{cases}\tag{7}$$

where

$$\Lambda\left(\mathbf{x},z,\mathbf{\Theta}\right)=p\left(c_i|D,\alpha\right)\frac{p\left(\mathbf{x}_{z(\mathbf{c})=k}|\mathbf{\Theta}\right)}{p\left(\mathbf{x}_{z(\mathbf{c})=m}|\mathbf{\Theta}\right)p\left(\mathbf{x}_{z(\mathbf{c})=l}|\mathbf{\Theta}\right)}\tag{8}$$

When Equation (7) converges to a stationary posterior, it provides number and regions of objects in an extracted segmentation blob.

V. Results

This section presents the real world data results of our proposed method for some challenging segmentation scenarios. A data set of the Karlsruhe Institute of Technology (KIT) is used for the experiments [23].

In order to estimate motion features from 3D point clouds, we first mapped the points on an occupancy grid as described in Section III. Grid cells have a resolution of 0.2 m in x and y dimensions. After removing the ground points, a connected components algorithm using 8 neighborhoods is applied to extract segmentation blobs. A threshold is set to apply a Kalman filter to the grid cells. If the height, length or width of a segmentation blob is more than 10 meter, motion vectors of grid cells in the blob are not estimated. This avoids tracking the grid cells of static structures such as buildings. The motion field estimation process is illustrated in Figure (1). For the Kalman filter from Section III-B, the state transition matrix $\mathbf{F}_t$ and the observation matrix $\mathbf{H}_t$ are given as

$$\mathbf{F}_t=\begin{bmatrix}1 & 0 & \triangle t & 0\\ 0 & 1 & 0 & \triangle t\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

where $\triangle t$ is the scan rate of the 3D LIDAR which is 0.1 seconds in our experiments.

$$\mathbf{H}_t=\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\end{bmatrix}$$

After selecting potentially moving segmentation blobs and estimating the motion features of their grid cells, the proposed MCMC based ddCRP is applied to evaluate its performance. The ddCRP explores the most likely partition structures in each segmentation blob. The estimated velocities in two dimensions are converted to one-dimensional movement directions as the feature vector for posterior inference. A window decay function $f(d)=1\left[d<a\right]$ is used with a value of $a=1$ to connect only adjacent grid cells. $F(\boldsymbol{\theta}_k)$ is a Gaussian distribution with $\boldsymbol{\theta}_k=\left(\mu_k,\sigma_k^2\right)$. Figure (2) illustrates the effect of the scaling parameter α on the cluster structure of a segmentation blob which contains two close objects.

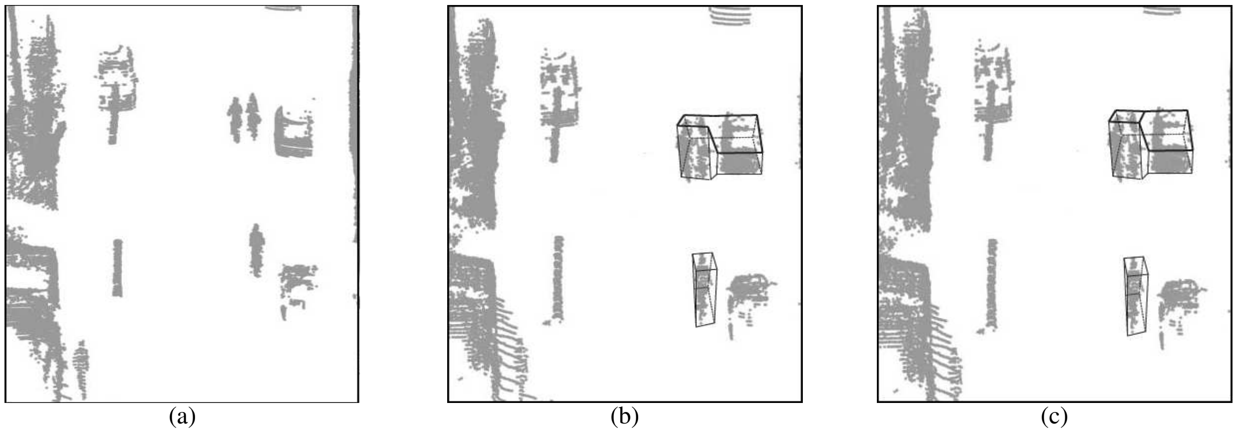


Fig. 5. (a) Two bicyclists are moving coherently (b) The extracted segmentation blob involves the two bicyclists and a parking car (c) Although the proposed method discriminates the car from the bicyclists, it fails to separate the two coherently moving bicyclists. A thick solid line shows the separation of the segmentation blob

The points represent the grid cells. For larger α values, the proposed algorithm intends to find non-robust smaller partitions, especially for grids where objects are close to each other. It is obviously seen in Figure (2) that a smaller scaling parameter α pushes the algorithm to larger and more robust clusters in segmentation blobs, so we set $\alpha = 10^{-4}$. The base distribution G_0 is a conjugate prior of the data generating distribution. Its parameters are $\mathbf{\Theta} = \{\mu_0, \sigma_0^2\}$. Although we independently run the ddCRP sampler with 30 iterations for each segmentation blob, it reached its stationary distribution within the first 10 iterations during our experiments.

Figure (3) depicts a scene where the assumption of well-segmented traffic participants does not hold. A bicyclist is getting closer to a parking car. The method described in Section III is performed to extract segmentation blobs by using spatial features and to estimate motion vectors of their corresponding grid cells. When two objects come close to each other, the algorithm extracts them in a segmentation blob as one object. That is shown in Figure (3) (c). For visualization, we fit the bounding box to the objects which we consider. Exploiting motion and spatial features together by using the proposed ddCRP based segmentation method, the bicyclist and the car are discriminated successfully. This can be seen in Figure (3) (d). The thick solid line in Figure (3) (d) represents the separation of the segmentation blob into two partitions which belong to the bicyclist and the parking car.

In the next example, a person is walking along a parking car as shown in Figure (4). Red dashed lines in Figure (4) (a) represent the trajectory of the person. When the segmentation algorithm which considers only spatial features is applied, the person can not be distinguished from the car and both objects are extracted in the same segmentation blob as given in Figure (4) (b). As our proposed method models the spatial correlation of objects, it benefits from the estimated motion features in the segmantation blob. Figure (4) (c) illustrates that the proposed ddCRP based segmentation method discerns the person in the blob.

In Figure (5) (a), two coherently moving bicyclists are displayed. When they get closer to each other and a parking car, their segmentation blob is extracted. It involves these three close objects which is illustrated in Figure (5) (b). Although our proposed method discriminates the car from the two bicyclists, it fails to separate the two bicyclists as shown in Figure (5) (c). As the spatial features are not sufficient for discrimination and the movement of the objects is coherent, the proposed method inherently miscarries. In order to handle this problem, some kind of appearence model can be integrated to the algortihm in addition to spatial and motion features, but this is subject to future work.

We note that specific tracking algorithms are not central to this paper. The center mass of points in the separated segmentation blobs and averaged velocities of their grid cells can be used for tracking approaches. Least mean square based spatial smoothing methods [14], [24], [25] can be applied to motion fields in order to obtain more accurate velocities for tracking stage.

VI. Conclusion

We have presented a novel segmentation method for close objects in 3D LIDAR point clouds. When objects get closer to each other, segmentation gets more difficult and often results under-segmentation in objects. The proposed method is exploiting the motion and spatial information simultaneously to discriminate close objects and to avoid the under-segmentation problem. In order to constitute motion features, a motion field of the surrounding environment is estimated from 3D LIDAR point clouds. A Markov chain Monte Carlo based distance dependent Chinese Restaurant Process framework is proposed to offer different segmentation hypotheses and decide on the most probable segments by exploiting motion and spatial features together.

The advantages of our proposed method are discussed for challenging real world scenarios. Using motion and spatial features together in the ddCRP framework produces successful segmentation results even when objects get very close to each other. The inherit limitation of our approach is that coherently moving objects can not be discriminated if they are spatially very close. In order to handle this problem, some kind of appearance model can be integrated to the algorithm in addition to spatial and motion features. The proposed method is not capable of running in real time due to the Markov chain inference. It would be interesting to apply a sequential Monte Carlo approach as a future work in order to turn the approach into a real time algorithm.

ACKNOWLEDGMENT

We acknowledge the support by the EU's Seventh Framework Programme under grant agreement no 607400 (TRAX, Training network on tRAcking in compleX sensor systems) http://www.trax.utwente.nl/

REFERENCES

[1] Velodyne LIDAR, *High Definition Lidar HDL-64E S2 Specifications*. Available: htpp://velodynelidar.com/lidar/hdlproducts/hdl64e.aspx.

[2] A. Petrovska and S. Thrun, *Model Based Vehicle Tracking for Autonomous Driving in Urban Environments*, The MIT Press: Zurich, Switzerland, 2008.

[3] J. Choi, S. Ulbrich, B. Lichte, and M. Maurer, *Multi-Target Tracking using a 3D-Lidar Sensor for Autonomous Vehicles*, In Proceedings of the 16th International IEEE Annual Conference on Intelligent Transportation Systems (ITSC 2013), The Hague, The Netherlands, October 6-9, 2013.

[4] A. Azim, and O. Aycard, *Detection, Classification and Tracking of Moving Objects on a 3D Environment*, Intelligent Vehicles Symposium, Alcala de Henares, Spain, June 3-7, 2012.

[5] P. Morton, B. Douillard, and J. Underwood, *An Evaluation of Dynamic Object Tracking with 3D Lidar*, In Proceedings of Australasian Conference on Robotics and Automation, Melbourne, Australia, 7-9 December, 2011.

[6] D. M. Blei and P. I. Frazier, *Distance Dependent Chinese Restaurant Processes*, Journal of Machine Learning Research, 12:24612488, August 2011.

[7] M. Himmelsbach and H. J. Wuensche,*Tracking and Classification of Arbitrary Objects with Bottom-Up Top-Down Detection*, Intelligent Vehicles Symposium, Alcala de Henares, Spain, 3-7 June, 2012.

[8] J. N. Driessen, and J. Biemond, *Motion Field Estimation for Complex Scenes*, Proc SPIE 1991, 1605, 511-521, 1991.

[9] A. Sellent, M. Eisemann, B. Goldlucke, D. Cremers, M. Magnor, *Motion Field Estimation from Alternate Exposure Images*, IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI), 33, 1577-1589. 2011

[10] F. Stein, *Efficient Computation of Optical Flow Using the Census Transform*, Proceedings of the 26th DAGM Symposium, pp.79–86, Lecture Notes in Computer Science, v.3175, 2004

[11] U. Franke, C. Rabe, H. Badino, and S. K. Gehrig, *6D Vision: Fusion of Stereo and Motion for Robust Environment Perception*, Proceedings of the 27th DAGM Symposium, pp.216-223, Vienna, Austria, August, 2005

[12] C. Rabe, U. Franke, S. Gehrig, *Fast Detection of Moving Objects in Complex Scenarios*, In proceedings of the IEEE Intelligent Vehicles Symposium, pp. 398-403, Istanbul, Turkey, 13-15 June, 2007.

[13] C. Rabe, T. Mller, A. Wedel, U. Franke, *Dense, Robust and Accurate 3D Motion Field Estimation from Stereo Image Sequences in Real-Time*, Proc. ECCV, Heraklion, Greece, 5.-11 Sept., 2010

[14] Q. Li, L. Zhang, Q. Mao, Q. Zou, P. Zhang, S. Feng, W. Ochieng, *Motion Field Estimation for a Dynamic Scene Using a 3D LIDAR*, Sensors, 14,16672-16691; doi:10.3390/s140916672, 2014

[15] J. Pitman, *Combinatorial Stochastic Process*, Lecture notes for St. Flour Summer School. Springer-Verlag, New York, NY, 2002

[16] T. S. Ferguson, *A Bayesian Analysis of Some Nonparametric Problems*, The Annals of Statistics, 1(2):209-230, 1973

[17] R. Socher, A. Maas, and C. D. Manning, *Spectral Chinese Restaurant Process: Nonparametric Clustering based on Similarities*, In Fourteenth International Conference on Artifical Intelligent and Statistics (AISTATS), 2012

[18] S. Ghosh, A. B. Ungureanu, E. B. Sudderth, and D. Blei, *Spatial Distance Dependent Chinese Restaurant Processes for Image Segmentation*, In NIPS, pages 1476-1484, 2011.

[19] S. Ghosh, E. Sudderth, M Loper, and M. Black, *From Deformations to Parts: Motion-based Segmentation of 3d Objects*, In Advances in Neural Information Processing Systems 25, pages 2006-2014, 2012.

[20] S. Geman, and D. Geman, *Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images*, IEEE Transactoins on Pattern analysis and Machine Intelligence, 6(6):721-741, November 1984

[21] C. Robert, and G. Casella, *Monte Carlo Statistical Methods*, Springer Texts in Statistics, Springer-Verlag, New York, NY, 2004

[22] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin, *Bayesian Data Analysis*, Texts in Statistical Science, Second Edition, Chapman and Hall CRC, 2003

[23] M. Frank, S. Christoph, *Velodyne SLAM*, In Proceedings of the IEEE Intelligent Vehicles Symposium, pp. 393-398, Baden-Baden, Germany, 5-9 June 2011

[24] J. Guo, J. Kim, *Adaptive Motion Vector Smoothing for Improving Side Information in Distributed Video Coding*, Journal of Information Processing Systems, Vol.7, No.1, March 2011

[25] L. Alparone, M. Barni, F. Bartolini, V. Capellini, *Adaptively Weighted Vector Median Filters for Motion Fields Smoothing*, In Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, pp.2267-2270, Atlanta, GA, USA, 7-10 May 1996