

A Comparison of Geometric and Energy-Based Point Cloud Semantic Segmentation Methods

Mathieu Dubois, Paola K. Rozo, Alexander Gepperth, Fabio A. González O., David Filliat

▶ **To cite this version:**

Mathieu Dubois, Paola K. Rozo, Alexander Gepperth, Fabio A. González O., David Filliat. A Comparison of Geometric and Energy-Based Point Cloud Semantic Segmentation Methods. 6th European Conference on Mobile Robotics (ECMR), IEEE, Sep 2013, Barcelona, Spain. pp.88-93, 10.1109/ECMR.2013.6698825 . hal-00963863

HAL Id: hal-00963863

https://inria.hal.science/hal-00963863

Submitted on 22 Mar 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A Comparison of Geometric and Energy-Based Point Cloud Semantic Segmentation Methods

Mathieu Dubois[1], Paola K. Rozo[2], Alexander Gepperth[1], Fabio A. González O.[2] and David Filliat[1]

***Abstract*— The recent availability of inexpensive RGB-D cameras, such as the Microsoft Kinect, has raised interest in the robotics community for point cloud segmentation. We are interested in the semantic segmentation task in which the goal is to find some relevant classes for navigation, wall, ground, objects, *etc.* Several effective solutions have been proposed, mainly based on the recursive decomposition of the point cloud into planes. We compare such a solution to a non-associative MRF method inspired by some recent work in computer vision. The MRF yields interesting results that are however less good than those of a carefully tuned geometric method. Nevertheless, MRF still has some advantages and we suggest some improvements.**

I. INTRODUCTION

The recent availability of inexpensive RGB-D cameras, such as the Microsoft Kinect, has made it possible to combine techniques developed for LIDAR and standard cameras for scene understanding (sometimes called image parsing). Therefore a lot of recent work in the computer vision community has been devoted to using RGB-D cameras for scene understanding while fewer works have studied the problem directly from a robotic point of view.

In the robotics community, recent years have seen the emergence of semantic mapping. The goal is to build robots able to autonomously map indoor environments and to recognize and localize significant elements such as objects, furnitures or rooms. RGB-D sensors are very interesting in this context because they give access to much more information than laser scans. However, object recognition, particularly at the category level, is a difficult task.

Therefore it is interesting to separate segmentation and recognition steps. In this article we present a system for the segmentation task: we try to recognize the structure of the environment (such as walls, ground), which is useful for navigation, and the presence of objects without identifying them. The precise identification of objects can be handled by a specialized module using a different set of techniques.

We present a system developed for semantic segmentation of RGB-D images based on a probabilistic framework to incorporate color, 3D shape and priors. This approach is similar to other energy minimization approaches found in the literature. We will compare the performance of our algorithm against a baseline approach purely based on the geometric segmentation of the point cloud [1].

[1]ENSTA ParisTech - INRIA Flowers Team 828 boulevard des Maréchaux 91762 Palaiseau CEDEX, France first.last@ensta.fr

[2] Universidad Nacional de Colombia, Of. 114 Edif. 453, Bogotá, Colombia {pkrozob, fagonzalezo}@unal.edu.co

Both methods have been developed in the context of the CAROTTE semantic mapping challenge[1]. In this challenge the robot explores an unknown environment and must provide a map containing as much information as possible. The environment covers approximately 120m^2 and contains several rooms typically 10 or more, with variable grounds and various difficulties (fitted carpet, tiling, grid, sand, stones). Several kinds of objects are present, either isolated or gathered, in multiple specimens, which must be detected, located, and identified by the robot. Figure 1 shows the robot and a typical environment.

The article is organized as follows. We review related works in section II and present our system in section III. Experiments are presented in section IV. Finally we conclude and present future works in section V.

II. STATE OF THE ART

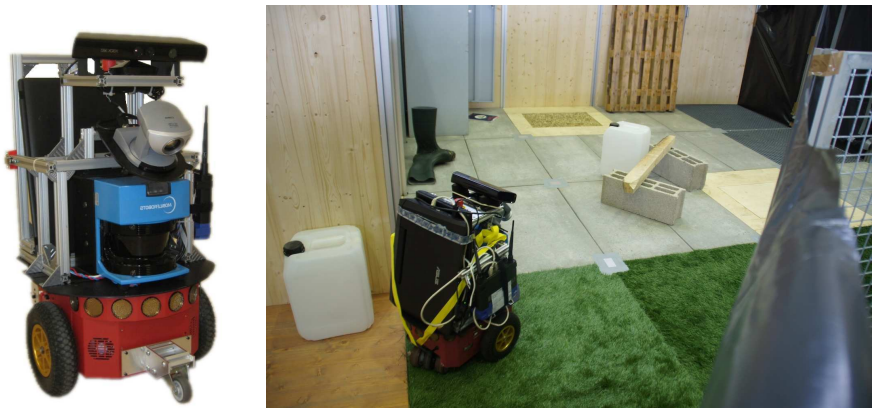
There is an extensive body of research in the field of 2D scene understanding and semantic segmentation [2], [3] and a comprehensive review is out of the scope of this paper. A lot of modern approaches [4], [5] use methods based on visual words [6]: the idea is to use an unsupervised vector quantization algorithm to find clusters (called visual words) in the set of image signatures. When a new image is presented the signatures are computed and matched to the closer cluster. This reduce the noise (because close signatures are mapped to the same word) and, because it acts as a form of discretization, allows to simplify learning.

There is also a lot of work on point cloud segmentation. Typical techniques are region growing methods [7], 3D Hough Transform [8] or ad-hoc methods exploiting domain knowledge [9] often based on decomposing the environment into planes.

Markov Random Field (MRF) [10] segmentation techniques have previously been applied to the point cloud segmentation and classification problem. Early works [11], [12], [13], [14] use Markov network that segments point clouds based on geometric features.

More recent works [15], [16], [17], [18] use similar models to incorporate color and 3D information and demonstrate that combining RGB and 3D information leads to superior results. Most of those works use first a bottom-up segmentation to simplify labeling. To model relationships between objects parts, it is necessary to use non-associative edge energies [14]. However, those studies are more oriented toward

[1]The CAROTTE competition is organized by the French research funding agency (ANR) and the French armament procurement agency (DGA). Website: http://www.defi-carotte.fr



(a) (b)

Fig. 1. Illustration of the CAROTTE challenge: (a) the robot we used is based on a Pioneer 3DX from Mobile Robots Inc., equiped with a Kinect to acquire RGB-D data (b) a typical room of the environment.

object recognition. Each paper use a different database and evaluation protocol so it's hard to compare MRF-based and ad-hoc methods for semantic segmentation ([17] compare their approach to a geometric approach for the task of support inference).

In the next section, we present a model combining visual words and MRF applied to semantic scene segmentation.

III. Using visual words and MRF for semantic segmentation

Our algorithm is composed of two main steps: we first segment the point cloud into N compact, homogeneous regions thanks to a bottom-up algorithm and then use a MRF model to assign a label to each region. Later, contiguous regions with the same label can be merged to form the final segments. The low-level algorithm is described in section III-A. The MRF model is described in section III-B and following. As the Kinect provides organized point clouds, each pixel in the 2D image corresponds to one point in the point cloud. In the following, we will use region and superpixel on one hand and point and pixel on the other hand, interchangeably.

A. Over-segmentation

For segmentation, we have adapted the SLIC superpixel algorithm [19] to take into account depth information. The algorithm allows to set the desired size of the regions S (and then the number of regions). The algorithm is an adaptation of the well-known K-means clustering algorithm to enforce compact regions: starting from initial seeds, at each step, points are attributed to the closest cluster and then clusters positions are recomputed as the average position of the points assigned in each cluster.

Given 2 pixels i and j, the distance between them $D(i,j)$ is:

$$D(i,j) = \sqrt{d_c(i,j)^2 + \frac{m^2}{S^2} d_s^2(i,j)} \quad (1)$$

where:

$$d_c(i,j) = \sqrt{(l_j - l_i)^2 + (a_j - a_i)^2 + (b_j - b_i)^2} \quad (2)$$

$$d_s(i,j) = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2 + (z_j - z_i)^2} \quad (3)$$

Where the z components are given by the depth map and $[l,a,b]^T$ is the color of the point in the Lab color space. The key difference with classical K-means is that the distance measure combines color and spatial proximity. As the expected size is controlled by the parameter S , the search region in the assignment step is limited to a neighborhood of size $2S \times 2S$. The parameter m allow to weight the relative importance between color similarity and spatial proximity: large values of m enforce compact superpixels while small values enforce adherence to image boundaries.

Compared to other bottom-up segmentation algorithms it takes into account both the color and the 3D information.

Depth maps obtained by structured light methods may contain missing values (due to infrared-absorbing surfaces, such as glass, or objects "shadow"). To remove those holes we use the method developed in [20] which uses a least-square interpolation on missing data.

B. MRF model

Following [15], we model the semantic segmentation problem in a probabilistic framework. We note $\Lambda = \{\ell_1, \ldots, \ell_m\}$ the set of classes (*i.e.* of possible label for each region). Given a point cloud segmented into N segments, we seek for the labeling $\overset{\star}{\mathbf{x}} = (x_1, \ldots, x_N)$ that satisfies:

$$\overset{\star}{\mathbf{x}} = \arg\max_{\mathbf{x}} \; \mathrm{P}(\mathbf{x}|\mathbf{A},\mathbf{S},\mathbf{G}) \quad (4)$$

where $\mathbf{A} = [\mathbf{a}_1, \ldots, \mathbf{a}_N]$ is the appearance features matrix, $\mathbf{S} = [\mathbf{s}_1, \ldots, \mathbf{s}_N]$ is the matrix of 3D shape descriptors and $\mathbf{G} = [\mathbf{g}_1, \ldots, \mathbf{g}_N]$ is the matrix of geometrical features.

We use vector quantization for appearance and shape features discretization. The matrix $\mathbf{A}$ is then approximated by the vector of appearance word $\mathbf{W}^{\mathbf{A}} = [w_1^A, \ldots, w_N^A]$ composed

of the index in the visual vocabulary. Similarly $\mathbf{S}$ is approximated by the vector of shape words $\mathbf{W^S} = [w_1^S, \ldots, w_N^S]$. Using Bayes' rule, Equation 4 can be written:

$$\begin{aligned} \overset{\star}{\mathbf{x}} &= \arg\max_{\mathbf{x}} \frac{\mathrm{P}\left(\mathbf{W^A}, \mathbf{W^S}, \mathbf{G}|\mathbf{x}\right)\mathrm{P}(\mathbf{x})}{\mathrm{P}(\mathbf{W^A}, \mathbf{W^S}, \mathbf{G})} \\ &= \arg\max_{\mathbf{x}} \mathrm{P}\left(\mathbf{W^A}, \mathbf{W^S}, \mathbf{G}|\mathbf{x}\right)\mathrm{P}(\mathbf{x}) \end{aligned} \tag{5}$$

The last line follows from the fact that $\mathrm{P}\left(\mathbf{W^A}, \mathbf{W^S}, \mathbf{G}\right)$ doesn't depend on $\mathbf{x}$. We assume that the geometrical features, the shape and the appearance features are independent conditionally to $\mathbf{x}$. Therefore:

$$\overset{\star}{\mathbf{x}} = \arg\max_{\mathbf{x}} \mathrm{P}\left(\mathbf{W^A}|\mathbf{x}\right)\mathrm{P}\left(\mathbf{W^S}|\mathbf{x}\right)\mathrm{P}(\mathbf{G}|\mathbf{x})\mathrm{P}(\mathbf{x}) \tag{6}$$

The appearance and shape features and the vocabulary construction are explained in section III-E.

The inference problem in Equation 6 is formulated in a binary MRF framework. The image is encoded as an undirected graph $\mathbf{G} = (\mathscr{V}, \mathscr{E})$ where the nodes $\mathscr{V}$ are the regions and the edges $\mathscr{E}$ are determined as follows: two nodes are connected if they share at least one boundary point.

Classically, we assume conditional independence between sites in the formulation of $\mathrm{P}\left(\mathbf{W^A}|\mathbf{x}\right)$, $\mathrm{P}\left(\mathbf{W^S}|\mathbf{x}\right)$ and $\mathrm{P}(\mathbf{G}|\mathbf{x})$. Note that [15] propose alternatives to this hypothesis.

In the MRF framework, the distribution $\mathrm{P}(\mathbf{x})$ can be factorized over the cliques of $\mathbf{G}$. Most studies use only binary cliques potential in $\mathrm{P}(\mathbf{x})$. However, [21] showed that unary terms (called external field) can be useful for segmentation especially when the class distribution is uneven which is true in our case. This term can be used to model the prior distribution of the classes.

The binary potential ensures spatial regularization but must be able to detect frontiers between classes. In the kind of environment we are interested in, object frontiers can be detected by an abrupt change in depth (*e.g.* boundary between an object and the wall behind it) or by a difference in normal orientation (*e.g.* between the ground and a wall).

Therefore we model $\mathrm{P}(\mathbf{x})$ as:

$$\begin{aligned} \mathrm{P}(\mathbf{x}) = \exp\Bigg(&-\sum_{i=1}^{N} \mathrm{g}_{\mathrm{i}}^{\mathrm{prior}}(x_i) - \sum_{(i,j)\in\mathscr{E}} \mathrm{g}_{\mathrm{ij}}^{\mathrm{depth}}(x_i, x_j) \\ &- \sum_{(i,j)\in\mathscr{E}} \mathrm{g}_{\mathrm{ij}}^{\mathrm{normals}}(x_i, x_j)\Bigg) \end{aligned} \tag{7}$$

The (unary) prior potential $\mathrm{g}^{\mathrm{prior}}$ is:

$$\mathrm{g}_{\mathrm{i}}^{\mathrm{prior}}(x_i) = -\log \mathrm{P}(x_i) \tag{8}$$

Following [15], the binary potential $\mathrm{g}^{\mathrm{depth}}$ is:

$$\mathrm{g}_{\mathrm{ij}}^{\mathrm{depth}}(x_i, x_j) = \begin{cases} 1 - e & \text{if } x_i = x_j \quad (9a) \\ \delta_{depth} + e & \text{else} \quad (9b) \end{cases}$$

where $e = \exp\left(-\frac{|\Delta z|^2}{2\sigma_{depth}^2}\right)$ (Δz is the depth difference between the 2 superpixels i and j).

To take into account the difference of normal orientation, the angle $\phi_{ij} \in [0, \pi]$ between the 2 normals is discretized into $K = 6$ bins ($\bar{\phi}_{ij} \in [\![0, K]\!]$ denotes the discretized angle). The binary potential $\mathrm{g}^{\mathrm{normals}}$ is then given by:

$$\mathrm{g}_{\mathrm{ij}}^{\mathrm{normals}}(x_i, x_j) = -\log \mathrm{P}\left(x_i, x_j|\bar{\phi}_{ij}\right) \tag{10}$$

Putting it all together, the optimal label is found by minimizing the energy:

$$\begin{aligned} E = &\lambda_{color}\sum_{i=1}^{N} E^A(i) + \lambda_{shape}\sum_{i=1}^{N} E^S(i) + \lambda_{geom}\sum_{i=1}^{N} E^G(i) \\ &+ \lambda_{prior}\sum_{i=1}^{N} E^{\mathrm{prior}}(i) + \lambda_{normals}\sum_{(i,j)\in E} E^{\mathrm{normals}}(i,j) \\ &+ \lambda_{depth}\sum_{(i,j)\in E} E^{\mathrm{depth}}(i,j) \end{aligned} \tag{11}$$

where:

$$E^A(i) = -\log \mathrm{P}\left(w_i^A|x_i\right) \tag{12}$$

$$E^S(i) = -\log \mathrm{P}\left(w_i^S|x_i\right) \tag{13}$$

$$E^G(i) = -\log \mathrm{P}(\mathbf{g_i}|x_i) \tag{14}$$

$$E^{\mathrm{prior}}(i) = -\log \mathrm{P}(x_i) \tag{15}$$

$$E^{\mathrm{normals}}(i,j) = \mathrm{g}_{\mathrm{ij}}^{\mathrm{normals}}(x_i, x_j) \tag{16}$$

$$E^{\mathrm{depth}}(i,j) = \mathrm{g}_{\mathrm{ij}}^{\mathrm{depth}}(x_i, x_j) \tag{17}$$

Note that the model is non-associative.

C. Learning

For a given class $\ell \in \Lambda$, learning the distributions $\mathrm{P}\left(w^A|\ell\right)$ and $\mathrm{P}\left(w^S|\ell\right)$ is very easy thanks to the discretization of appearance and shape descriptors. Similarly the discrete probability distribution $\mathrm{P}\left(\ell, \ell'|\bar{\phi}\right)$ and the prior distribution $\mathrm{P}(\ell)$ are estimated on the training set. We use the Laplace method to avoid null probabilities.

For geometric features, we consider the position in the 2D image $\mathbf{g} = [x, y]^T$. Given the fact that the robot navigates, the x position of an object may not be relevant. To take this into account, we use the distance to the center of the image: the coordinate x is mapped in the range $[0, W/2]$ (where W is the image width). The distribution $\mathrm{P}(\mathbf{g}|\ell)$ is approximated by a two-dimensional Gaussian distribution:

$$\mathrm{P}(\mathbf{g}|\ell) = \frac{1}{2\pi\sqrt{|\Sigma_\ell|}} \exp\left(-\frac{1}{2}(\mathbf{g} - \mu_\ell)^T \Sigma_\ell^{-1}(\mathbf{g} - \mu_\ell)\right)$$

We therefore have to evaluate 5 parameters per class: the mean positions μ_ℓ^x and μ_ℓ^y, the variances σ_ℓ^x and σ_ℓ^y and the correlation between the coordinates ρ_ℓ.

The parameters of $\mathrm{g}^{\mathrm{depth}}$ and the weights are manually set (see section IV).

D. Inference

To solve the inference problem, we use the approach of [22] which formulates a relaxed version of the problem as a linear program. A specific algorithm is designed to efficiently solve this problem.

Because it is a relaxed version of the problem, a crisp class is not attributed to each site. Instead, for a given node $i \in \mathcal{V}$, every class ℓ receives a confidence value $0 \leq \varphi_i(\ell) \leq 1$. Usually the class with the larger value φ_i is attributed to the region. However, using the distribution of φ_i can be useful to detect regions which are not well classified for instance because they correspond to a previously unseen object. As learning is incremental the robot can then focus on the unknown regions and update the probability distributions used in the model.

E. Features and visual words

For each region, the appearance features are computed by concatenating the 128-dimensional SIFT features [23] (computed at the morphological center of each region) and the average Lab values on the region to give a 131-dimensional feature vector. For shape features, we use the 496-dimensional features of [24] (also computed at the morphological center).

The visual and shape vocabularies are trained using the optimized learning rate LVQ1 procedure [25] (see section IV).

IV. EXPERIMENTS

A. Database

We use a database of point clouds acquired during the CAROTTE challenge. The point clouds where acquired while the robot moves autonomously as described in [1]. This avoid the potential bias in databases acquired by manually guided robots or cameras as in [18].

The database consists of 100 point cloud with high accuracy annotations: each pixel in the RGB image is assigned a class. Care was taken to perform an appearance-based labeling which means that parts of, e.g., a wall which are visually strongly different will be assigned different identities. This was done in order to permit appearance-based segmentation and is interesting in the context of the CAROTTE challenge. In this article, all types of ground are counted as a single "ground" class, and similarly for wall and object types which together form the "wall" and "object" classes[2]. Figure 2 shows an example of image from the database. Due to the structured light method, the depth can't be measured on some border pixels of the RGB image (this is a different problem than the missing data in section III-A). The RGB image provided by the Kinect is cropped to 410×560 pixels (starting at position $(50, 40)$) to remove border pixels.

B. Baseline system

We compare our system to the one described in [1] which is based on geometric segmentation using only distance information. This system is inspired by [9], doesn't require training but has several parameters which were manually tuned. It showed interesting performances during the CAROTTE competition.

[2]The database can be freely downloaded at `http://cogrob.ensta.fr/pacom`

The point cloud is filtered to remove distant outliers, and re-sampled using a voxel grid to have a constant spatial resolution (2 cm).

An off-line calibration phase, is used to detect the ground plane in an ideal setting using RANSAC [26]. We assume that the coefficients in its defining equation stay roughly constant wherever the robot goes. During segmentation, this equation is used to detect the ground: surface normals are calculated for each point in the point cloud and all points sufficiently close to the theoretical ground plane, and having a normal pointing upwards, are recognized as part of the ground and removed from the cloud.

Next walls are iteratively detected using RANSAC. A wall is defined as a plane of sufficient size with normals parallel to the ground plane. Points on and close to detected walls are removed as well, and the procedure is repeated until no more walls are detected.

The remaining points are then projected onto the theoretical ground plane and subsequently grouped according to their (projected) distance. For this purpose, we use a region-growing type of algorithm to form groups of 3D points that may correspond to objects. These groups form the initial segmentation results. To segment objects placed on planar surfaces of other objects, such as desks, tables or shelves, we continue to analyze the point cloud that is left after floor and wall removal to detect horizontal planes, and we remove all points below this level. The remaining points are grouped using the same region-growing clustering method as before, leading to additional segmentation results. In the event that several horizontal planes are detected (for example if the image contains two shelves), this process is iterated from the highest to the lowest detected horizontal plane, always producing segmentation results from the points above the current horizontal plane, and repeating the whole step for all the points below it. Figure 3 shows an example of segmentation with the baseline algorithm.

Due to the fact that the system don't use color information, it is unable to detect object smaller than a few centimeters high on the ground which may block the robot. The geometric model was manually tuned and is used as a black box here.

C. Results

For superpixels, we set $S = 15 pixels$ and $m = 10$. The vocabularies are trained on a random subsample of the features (using 30 % of the features). We use 200 words for the color and depth vocabularies.

We have run different experiments in order to assess the influence of the non-learned parameters. In all experiments, the values of δ_{depth} and σ_{depth} (for computing g^{depth}) are fixed (respectively to 0.8 and 0.05). Similarly, the weights λ_{color}, λ_{shape}, λ_{geom} and λ_{prior} are set to 1.0.

Regularization weights λ_{normal} and λ_{depth} are always equal and can take 3 different values: 0 (*i.e.* no regularization), 0.2 (*i.e.* weak regularization) and 1.0 (*i.e.* strong regularization). For each value, we run a 5-fold cross validation procedure



(a) RGB image (b) Labeled image

Fig. 2. Examples of image annotations in the PACOM RGBD segmentation database. (a): Original image. (b): annotated image. Each pixel has an identity associated to it, where different pixel colors indicate different identities.



Fig. 3. Example of the segmentation with the baseline system. Left: original point cloud after re-sampling. Right (upper row): detected walls which are subsequently removed. Right (lower row): Segmented objects including objects placed on top of each other. The whole group is segmented as a single object at first. Detection of the horizontal place on the metal suitcase allows to separate the suitcase from the three objects placed on top of it.

on the database to evaluate the performances. The evaluation metrics are the pixel-wise precision and recall.

Table I summarizes the results. The performances of the MRF on the "wall" class is comparable to the baseline (precision is better but recall a bit less good). For the "ground" class, results are less good but still rather high. For those 2 classes (which are well represented in the training set), the influence of the regularization weights is not very important.

However for the class "objects", the MRF model lags behind the baseline in particular in terms of recall. Several hypotheses can be made to explain this fact. First, the object class is far more diverse in shape and appearance and thus harder for a learning based method. Second, our regularization terms might not capture enough relation between classes. In particular, objects and ground tend to have similar angles than ground and walls. Therefore the MRF tend to label as objects as "wall" (because this class is far more frequent).

V. CONCLUSION AND FUTURE WORKS

A. *Conclusion*

The results shows that the geometric method gives superior results for the task of semantic segmentation in particular for the object class. This can be explained by the fact that it incorporates a lot of domain knowledge (namely that indoor environments are made of planes and that objects lie on top of them).

However, MRF segmentation gives interesting results and has several advantages. First most of it's components can be used for other purpose or in other, less constrained, environments where domain knowledge is not available. For instance we could try to recognize more precisely the objects. Second it requires less tuning since most parameters are learned from the database. Third, it uses the appearance information which

Algorithm	Precision			Recall			Overall
	Walls	Ground	Objects	Walls	Ground	Objects	
Baseline	93.3	97.8	65.0	87.7	91.2	98.1	94.9
MRF strong regul.	96.4	89.5	46.8	77.6	79.5	41.6	76.0
MRF weak regul.	94.7	89.4	88.1	82.8	80.4	23.5	77.86
MRF no regul	94.7	88.8	64.8	81.1	78.5	32.1	76.8

TABLE I

RESULTS OF THE SEMANTIC SEGMENTATION TASK. "BASELINE" STANDS FOR THE GEOMETRIC MODEL FROM [1]. MRF STANDS FOR THE ENERGY-BASED MODEL.

could help to identify different types of ground or wall (this was one of the goal in the CAROTTE challenge). Last but not least, as it gives a probabilistic output, it allows the robot to draw hypothesis on the environment and adapt its behavior. Therefore we think it is interesting to investigate improvements.

B. Future works

For now our algorithm doesn't fully exploit the structure of the point clouds. For instance, the SLIC segmentation procedure could be modified to use the real distances in meters in the distance computation and search in the full 3D neighborhood. Similarly the neighborhood and the geometric positions of the classes could be computed with the real-world coordinates. As we have mentioned, it seems necessary to investigate more complex edge potentials. Finally more investigations could be done on learning the parameters.

REFERENCES

[1] D. Filliat, E. Battesti, S. Bazeille, G. Duceux, A. Gepperth, L. Harrath, I. Jebari, R. Pereira, A. Tapus, C. Meyer, S.-H. Ieng, R. Benosman, E. Cizeron, J.-C. Mamanna, and B. Pothier, "RGBD object recognition and visual texture classification for indoor semantic mapping," in *Proceedings of the 4th International Conference on Technologies for Practical Robot Applications (TePRA)*, 2012, pp. 127 – 132. [Online]. Available: http://hal.archives-ouvertes.fr/hal-00755295

[2] L.-J. Li, R. Socher, and L. Fei-Fei, "Towards total scene understanding: Classification, annotation and segmentation in an automatic framework," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, vol. 2, IEEE. Miami: IEEE Computer Society, June 2009, pp. 2036–2043.

[3] K. Murphy, A. Torralba, D. Eaton, and W. Freeman, "Object detection and localization using local and global features," *Lecture Notes in Computer Science*, 2005.

[4] E. B. Sudderth, A. Torralba, W. T. Freeman, and A. S. Willsky, "Learning hierarchical models of scenes, objects, and parts," in *International Conference on Computer Vision*, vol. 2, 2005, p. 331–1338.

[5] J. Tighe and S. Lazebnik, "Superparsing - scalable nonparametric image parsing with superpixels," *International Journal of Computer Vision*, vol. 101, no. 2, pp. 329–349, 2013.

[6] G. Csurka, C. Bray, C. Dance, and L. Fan, "Visual categorization with bags of keypoints," in *Workshop on Statistical Learning in Computer Vision, ECCV*, 2004, pp. 1–22.

[7] X. Y. Jiang and H. Bunke, "Robust and fast edge detection and description in range images," in *MVA*, 1996, pp. 538–541.

[8] T. Rabbani and F. Van Den Heuvel, "Efficient Hough transform for automatic detection of cylinders in point clouds," *ISPRS WG III/3, III/4*, vol. 3, pp. 60–65, 2005.

[9] R. B. Rusu, N. Blodow, Z. C. Marton, and M. Beetz, "Close-range scene segmentation and reconstruction of 3D point cloud maps for mobile manipulation in domestic environments," in *IROS*. IEEE, 2009, pp. 1–6.

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

[11] D. Anguelov, B. Taskar, V. Chatalbashev, D. Koller, D. Gupta, G. Heitz, and A. Y. Ng, "Discriminative learning of Markov random fields for segmentation of 3D scan data," in *CVPR (2)*. IEEE Computer Society, 2005, pp. 169–176.

[12] R. Triebel, R. Schmidt, Ó. M. Mozos, and W. Burgard, "Instance-based AMN classification for improved object recognition in 2D and 3D laser range data," in *IJCAI*, M. M. Veloso, Ed., 2007, pp. 2225–2230.

[13] E. H. Lim and D. Suter, "3D terrestrial LIDAR classifications with super-voxels and multi-scale conditional random fields," *Computer-Aided Design*, vol. 41, no. 10, pp. 701–710, 2009.

[14] R. Shapovalov, A. Velizhev, and O. Barinova, "Non-associative Markov networks for 3D point cloud classification," *Photogrammetric Computer Vision and Image Analysis*, 2010.

[15] B. Micusik and J. Kosecka, "Semantic segmentation of street scenes by superpixel co-occurrence and 3D geometry," in *Proccedings of IEEE 12th International Conference on Computer Vision, ICCV 2009*. IEEE, 2009.

[16] A. Collet, S. S., and M. Hebert, "Structure discovery in multi-modal data: a region-based approach," in *ICRA*. IEEE, 2009.

[17] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from RGBD images," in *ECCV (5)*, ser. Lecture Notes in Computer Science, A. W. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, Eds., vol. 7576. Springer, 2012, pp. 746–760.

[18] A. Anand, H. S. Koppula, T. Joachims, and A. Saxena, "Contextually guided semantic labeling and search for three-dimensional point clouds," *International Journal of Robotics Research*, vol. 32, no. 1, pp. 19–34, 2013.

[19] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2012.

[20] A. Janoch, S. Karayev, Y. Jia, J. Barron, M. Fritz, K. Saenko, and T. Darrell, "A category-level 3-D object dataset: Putting the kinect to work," ICCV Workshop on Consumer Depth Cameras in Computer Vision, 2011.

[21] G. Celeux, F. Forbes, and N. Peyrard, "EM-based image segmentation using Potts models with external field," in *RFIA*, 2004.

[22] T. Werner, "A linear programming approach to max-sum problem: A review," *IEEE PAMI*, vol. 29, no. 7, pp. 1165–1179, 2007.

[23] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[24] J. Shotton, A. W. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, "Real-time human pose recognition in parts from single depth images," in *CVPR*. IEEE, 2011, pp. 1297–1304.

[25] T. Kohonen, "New developments of Learning Vector Quantization and the Self-Organizing Map," in *Symp. on Neural Networks; Alliances and Perspectives in Senri*. Osaka, Japan: Senri Int. Information Institute, 1992.

[26] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.