In Proceedings of the 4th International Conference on Intelligent Robotics and Applications (ICIRA), Aachen, Germany, 2011

# Efficient Multi-Resolution Plane Segmentation of 3D Point Clouds

Bastian Oehler[1], Joerg Stueckler[2], Jochen Welle[1], Dirk Schulz[1], and Sven Behnke[2]

[1] Research Group Unmanned Systems, Fraunhofer-Institute for Communication, Information Processing and Ergonomics (FKIE), 53343 Wachtberg, Germany
[2] Computer Science Institute VI, Autonomous Intelligent Systems (AIS), University of Bonn, 53012 Bonn, Germany

**Abstract.** We present an efficient multi-resolution approach to segment a 3D point cloud into planar components. In order to gain efficiency, we process large point clouds iteratively from coarse to fine 3D resolutions: At each resolution, we rapidly extract surface normals to describe surface elements (surfels). We group surfels that cannot be associated with planes from coarser resolutions into co-planar clusters with the Hough transform. We then extract connected components on these clusters and determine a best plane fit through RANSAC. Finally, we merge plane segments and refine the segmentation on the finest resolution. In experiments, we demonstrate the efficiency and quality of our method and compare it to other state-of-the-art approaches.

**Keywords:** Plane segmentation, multi-resolution, RANSAC, Hough transform

## 1 Introduction

Depth sensors such as 3D laser range finders or the Microsoft Kinect provide dense 3D measurements that typically consist of millions of points. In robotics applications like object manipulation or teleoperation, it is often crucial to interpret this massive amount of data in real-time. In this paper, we propose efficient means to segment 3D point clouds into planar segments (see Fig. 1). In structured environments, the planar segmentation gives a compact representation of scene content. It can also be used to generate object hypotheses by focussing on point clusters not explained by the main planes.

We gain the efficiency required for the processing of large point clouds by adopting a coarse-to-fine strategy: We extract surface elements (surfels, described by location, extent, and surface normal) on multiple resolutions, starting from the coarsest one. On each resolution, we associate the surfels with planar segments that have been found on coarser resolutions. New planes are created from the remaining unassociated surfels by first grouping them into co-planar clusters with the Hough transform. We split each cluster into a set of connected components. For each connected component, we apply RANSAC to determine a best plane fit and to reject outliers robustly. In a final processing step, we merge plane segments and refine the segmentation on the finest resolution.

The use of a coarse-to-fine strategy has several advantages over a segmentation on a single resolution. Firstly, large plane segments can be detected from only few surfels

Source code: https://github.com/fkie/structure_coloring_fkie

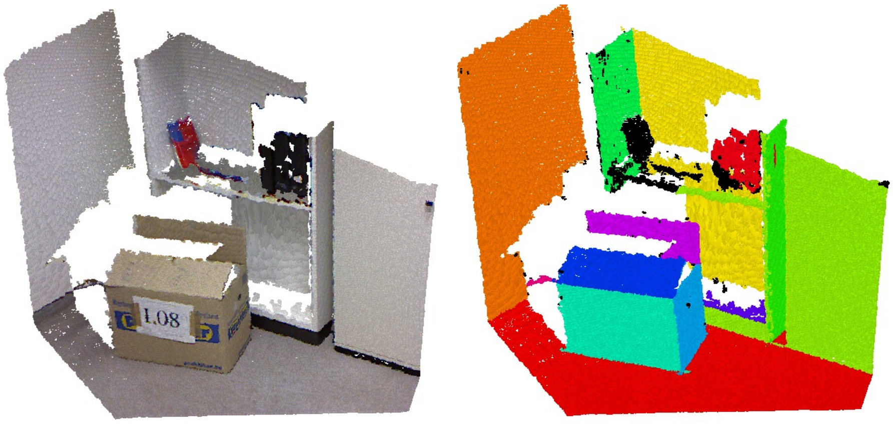

**Fig. 1.** Example scene and its planar segmentation.

which renders our method very efficient. Furthermore, our approach handles variability in the extent of plane segments more robustly, since it uses as much context as possible to decide for co-planarity.

We organize this paper as follows: After a brief discussion of related work in Sec. 2, we detail our approach to plane segmentation in Sec. 3. Finally, we evaluate our method in experiments in Sec. 4.

## 2 Related Work

The robotics and computer graphics communities have developed a rich set of approaches for segmenting a scene into geometric shape primitives. Many of these approaches also extract plane segments. We identify three main lines of research: approaches based on Random Sample Consensus (RANSAC, [1]), methods using the Hough transform, and algorithms that perform region growing on depth images.

RANSAC [1] is a method to robustly fit a model into a set of data points that may contain even a large number of outliers. It randomly selects a minimal set of data points for estimating the model parameters. From the random samples, it chooses the one that is best supported by the complete set of points. As of its general formulation, RANSAC can be easily applied to fit any kind of geometric shape primitive. However, the basic RANSAC approach assumes that only one model can be fit to the data. Schnabel et al. [10] propose to extend basic RANSAC through multi-resolution and locality heuristics. Instead of uniformly drawing samples for the minimal sample set from the complete point cloud, they sample from a local normal distribution in 3D Cartesian space. Furthermore, they randomly select a sampling scale in an octree representation of the point cloud. By such means, they achieve an efficient and robust method. Gotardo et al. [3] extract planar surfaces with a modified RANSAC approach. They first extract an edge map from the depth gradient image and determine connected components in the edge map. Then they apply RANSAC to robustly fit planar segments into the connected components.

The Hough transform [6] is an alternative approach to estimate model parameters from a set of measurements. In contrast to RANSAC, its formulation is sound when the measured points support multiple instances of the model with different parametrizations (such as different planes in a scene). It transforms the given measurements from the original space (e. g., $\mathbb{R}^3$ for point clouds) into a set of possible parameter vectors in the parameter space of the model. Each original point votes for a manifold in parameter space. Vote clusters in the parameter space then represent the model fits. The Hough transform is computationally demanding when extracting geometric shape primitives from 3D point clouds, because such models contain multiple parameters which corresponds to a high-dimensional Hough space. Consequently, in a naive implementation a single point in 3D must vote for a high-dimensional manifold in parameter space. The quality of the model fit strongly depends on the clustering method in parameter space. Using histograms, for example, the quality is affected by the discretization of the histogram. Vosselman et al. [12] propose an efficient variant of the Hough transform for extracting shape primitives. They estimate model parameters from surfels (points with local surface normals) and divide model fitting into several stages. In a first stage, they apply the Hough transform to find parallel surfels. Secondly, they find co-planar surfels in parallel surfels with a similar distance to the origin of the coordinate frame. In our approach, we improve this method in efficiency and accuracy by applying a coarse-to-fine strategy and combining the Hough transform with efficient RANSAC.

Both, the Hough transform and RANSAC, are global methods that ignore point neighborhood. Some methods have been proposed that exploit the neighborhood information in dense depth images. In [7] or [2], region-growing and region-merging techniques are applied to extract planar segments. Taylor et al. [11] use a split-and-merge approach. Finally, Harati et al. [4] extract edges from bearing-angle images to find connected components. Our approach is not restricted to point clouds for which an image-like point neighborhood is known. It can be readily applied to point clouds that have been registered from multiple views. We consider the neighborhood of points, since we process surface normals in local neighborhoods on multiple resolutions. In addition, we reestablish the neighborhood of surfels by finding connected components in co-planar surfels.

## 3 Efficient Multi-Resolution Segmentation into Planar Segments

We combine the Hough transform with RANSAC to robustly extract plane segments from 3D point clouds (s. Fig. 2). In order to improve efficiency, we use a coarse-to-fine strategy: We extract local surface normals at multiple resolutions to describe surface elements (surfels). We implemented a highly efficient method for multi-resolution normal estimation using octrees. At each resolution, we determine which surfels can already be explained by planes that have been found on coarser resolutions. On the remaining surfels, we apply the Hough transform to pre-segment the scene into co-planar surfels. In order to improve accuracy and robustness, we fit plane segments on connected components using RANSAC. On the finest resolution, we merge co-planar connected plane segments and distribute the remaining points.

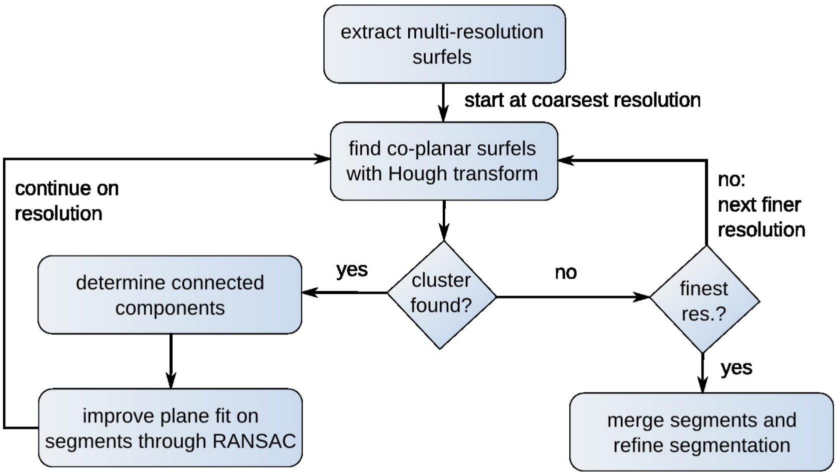
extract multi-resolution surfels
start at coarsest resolution
find co-planar surfels with Hough transform
continue on resolution
no: next finer resolution
determine connected components
yes
cluster found?
no
finest res.?
yes
improve plane fit on segments through RANSAC
merge segments and refine segmentation

**Fig. 2.** Overview over our coarse-to-fine planar segmentation approach. See text for details.

### 3.1 Efficient Normal Estimation through Multiple Resolutions

We represent the point cloud with an octree. An octree consists of branching nodes and leaf nodes that each cover a 3D volume. The root of the tree spans the complete 3D volume of interest. Each branching node divides its volume into eight equally sized cubes (called octants) at its center position. For each of its octants, the node contains a child that is either a branching node itself or a leaf in the tree.

The octree can naturally be used to regularly sample the point cloud at sampling resolutions that correspond to the volume sizes of the nodes in the various depths of the tree. For a sampling depth $d$, we determine all nodes at the sampling depth or all leaf nodes at coarser resolutions. Furthermore, the octree allows to efficiently calculate integral values in the volume of nodes: In each node, we maintain the integral over the values of points that lie within the volume of the node. During the construction of the tree, we distribute the value of a point to all nodes that the point visits while it is recursively passed from the root to its final leaf node.

We exploit this property to efficiently calculate the mean and sample covariance of points in each node. For the mean $\boldsymbol{\mu} = \frac{1}{N}\sum_i \boldsymbol{p}_i$, we simply maintain the number of points $N$ and the sum of 3D point coordinates $\sum_i \boldsymbol{p}_i$. The sample covariance can be obtained by the computational formula of variance

$$Cov(\boldsymbol{p}) = E\left[\boldsymbol{p}\boldsymbol{p}^T\right] - \left(E\left[\boldsymbol{p}\right]\right)\left(E\left[\boldsymbol{p}\right]\right)^T = \frac{1}{N}\sum_i \boldsymbol{p}_i\boldsymbol{p}_i^T - \boldsymbol{\mu}\boldsymbol{\mu}^T.$$

Thus, it suffices to additionally maintain the sum of squared 3D point coordinates $\sum_i \boldsymbol{p}_i\boldsymbol{p}_i^T$ in each node.

Once the octree has been constructed, we estimate surface normals at every resolution by finding the eigenvector to the smallest eigenvalue $\lambda_0$ of the sample covariance ($\lambda_0 \leq \lambda_1 \leq \lambda_2$). Fig. 3 shows example normals extracted on two resolutions with our method.

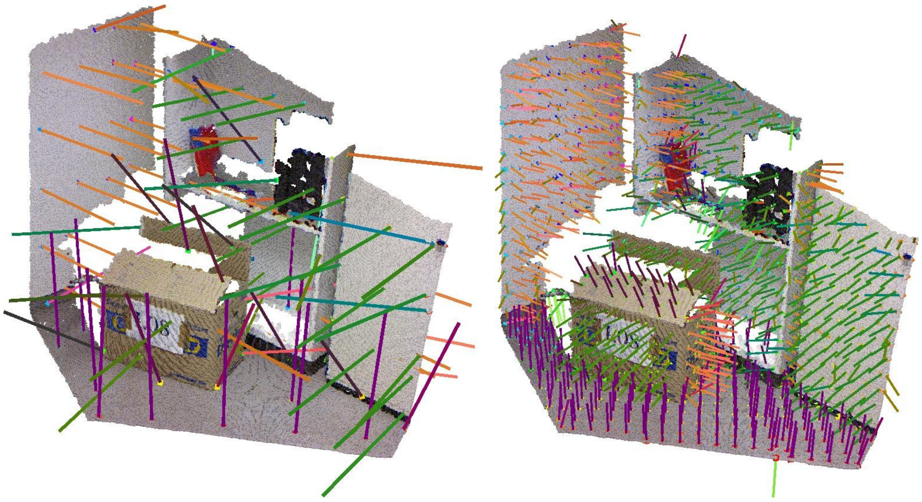

**Fig. 3.** Normals on a coarse (left) and a fine (right) resolution estimated with our fast normal estimation method. The normals are color-coded for direction (best viewed in color). The length of the normals indicates the space discretization.

We require a minimum number of points to support the normal estimate. We further evaluate the quality of the normals by considering the eigenvalues of the covariance matrix. The curvature

$$\gamma := \frac{\lambda_0}{\lambda_0 + \lambda_1 + \lambda_2} \tag{1}$$

is small, when the points are mainly distributed in a plane. The relation $\lambda_1/\lambda_2$ is close to one, when the points are equally distributed in both directions in the plane. Thresholding these indicators rejects normal estimates on ridges and at plane borders. Finally, we require the largest eigenvalue $\lambda_2$ to be large in relation to the volume length of the node, such that the points fully spread the volume size.

### 3.2 Pre-segmentation in Hough-Space

We find clusters of co-planar surfels on a single resolution with the Hough transform. Similar to Vosselman et al. [12], we use a fast two-stage approach.

In the first stage, each surfel votes for planes with corresponding normals in an orientation histogram. We discretize the orientation histogram approximately equidistant in inclination and azimuth angles following the approach by Rabbani [8]. The curvature $\gamma$ in Eq. (1) provides a measure of uncertainty in the normal estimates. We use this curvature to distribute the normal orientation of surfel $k$ with a weight $w_j$ to a range of bins $j$ with similar orientations in the histogram, i. e.,

$$w_j = N_k \cdot \left(1 - \frac{\gamma_k}{\gamma_{\max}}\right) \cdot \left(|\langle \boldsymbol{n}_k, \boldsymbol{n}_j \rangle| - \cos(\alpha)\right) / \cos(\alpha),$$

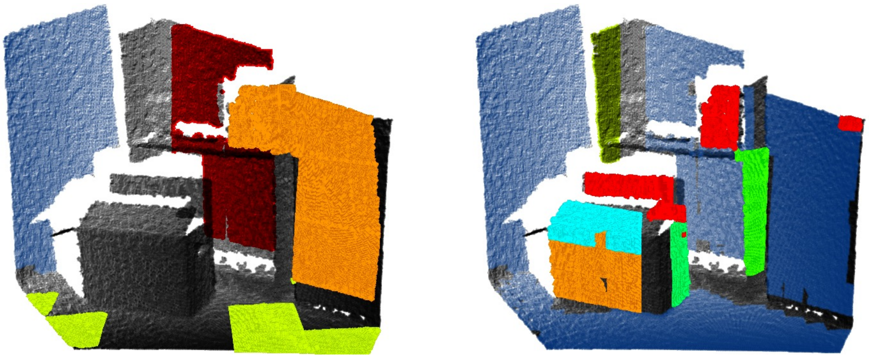

**Fig. 4.** Clusters found by the Hough transform on two consecutive resolutions (grey: not yet segmented, blue: previously segmented).

where $N_k$ is the number of points in the surfel, $\gamma_{\max}$ is the threshold on the normal curvature, $n_k$ and $n_j$ are the normal of the surfel and the histogram bin, respectively, and $\alpha < 90°$ is the angular influence range of the normal. We detect local maxima in the orientation histogram in order to find clusters of parallel surfels.

In the second stage, we determine co-planar surfels from the clusters of parallel surfels. Each surfel votes for the distance of a plane to the origin of the coordinate frame (e. g., the view point). Similar to the orientation histogram, we distribute votes onto neighboring bins with a linear decay. We find clusters of co-planar surfels again at the maxima of the distance histogram. Figure 4 shows results of this pre-segmentation step in our example scene. To make this process efficient, we keep the resolution of the histograms coarse and postpone an accurate estimate of the model parameters to later processing stages. The resolution of the distance histograms is increased with the resolution of the surfels, however.

### 3.3 Segmentation into Connected Components

The Hough transform does not consider the spatial connectivity of surfels. We therefore extract connected components from the sets of co-planar surfels. Fig. 5 illustrates this with an example. We overlay a grid over the plane that corresponds to the Hough-space maximum of the surfels. The resolution of the grid is chosen according to the resolution of the surfels. We project each surfel position into the grid and mark occupied grid cells. Region-growing yields connected components which we discard when the component is not supported by a minimum number of surfels (set to 3 in our implementation).

### 3.4 Accurate Segmentation through RANSAC

We further improve the plane fit to the connected components of co-planar surfels. The plane estimate by the Hough transform is only a rough estimate of the true underlying plane due to the coarse resolution of the orientation and distance histograms. We therefore apply RANSAC directly to the points represented by the surfels. An example of the outlier detection is visualized in Fig. 6.

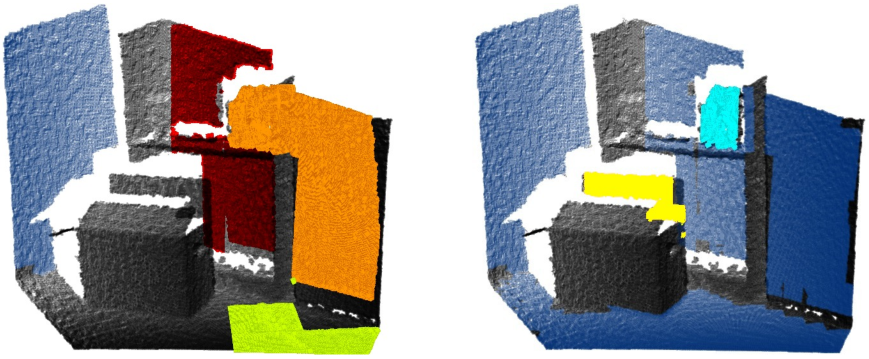

**Fig. 5.** Segmentation of the clusters from Fig. 4 into connected components. In the left image, the green segment is split into two components. The left component is not supported by sufficient surfels and is discarded. In the right image, the segmentation is only shown for the red cluster from Fig. 4, which is split into two planar segments (yellow and cyan). The rightmost part is discarded for low support.

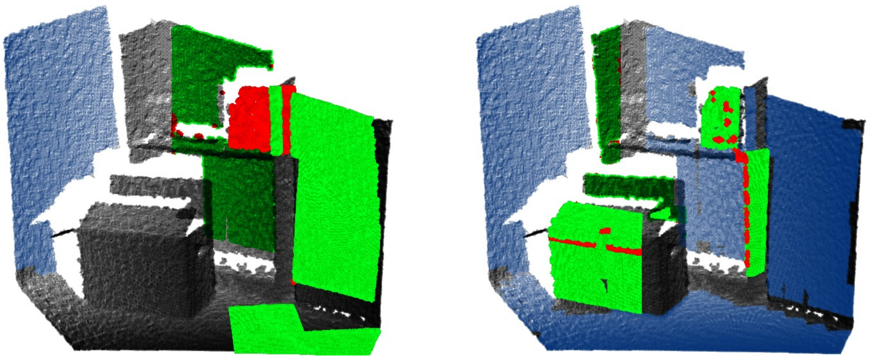

**Fig. 6.** RANSAC refinement on connected components (green: inliers, red: outliers).

RANSAC estimates plane parameters from a random set of three point samples. Within a fixed number of iterations, we determine the plane estimate that is best supported by all points of the surfels. Points are accepted as inliers to the plane fit when their distance to the plane is below some threshold. We adapt this threshold to the resolution of the surfels. We only accept plane fits that are supported by a large fraction of the surfel points. We also require the extracted plane to be similar to the initial fit determined by the Hough transform. When the plane fit is accepted, we redetermine the connected component of the segment.

### 3.5 Coarse-To-Fine Segmentation

In the previous sections we detailed how we segment planes on a single resolution. We propose however to segment a scene with a coarse-to-fine strategy. By this, large plane segments can be detected efficiently from only few surfels. Furthermore, our approach inherently adapts to the extent of planes in the scene. It uses as much context as possible to decide for co-planarity.

We process the scene from coarse to fine resolutions. We transit to the next finer resolution, when no more plane segments are found on a resolution. In order to improve the segmentation of the already found plane segments, we redistribute the surfels on the finer resolution onto the segments. We test, if the surfel orientation and position fits well to each plane segment, and if it lies within or at the border of its connected component.

Eventually, we also adapt the connected components. For this purpose, we increase the sampling rate of the occupancy map according to the new resolution. We project the surfels into the plane segment and mark the corresponding cells occupied. However, we keep the coarser occupancy decisions from previous layers. Note, that while plane segments may expand during this process, segments that grow together are not merged. We merge co-planar connected segments in a final processing step.

### 3.6 Post-Processing

After all resolutions have been processed, we improve the segmentation on the finest resolution. First, we merge connected co-planar plane segments. We then distribute the nodes onto the plane segments without using normal information. For each node, we determine a list of plane segment candidates with small distance towards the mean of the points within the node's volume. In addition, the node needs to fall within the connected component or at the borders of each candidate.

We further examine nodes that could not be assigned uniquely to plane segments and distribute the points in the node's volume individually. We pick the best two plane segment candidates $s_1, s_2$ according to distance and compute the equidistal plane through the intersecting line of $s_1$ and $s_2$ with normal direction

$$\boldsymbol{n}_{cut} = \left( \frac{\boldsymbol{n}_{s_1} + \boldsymbol{n}_{s_2}}{2} \right) \times (\boldsymbol{n}_{s_1} \times \boldsymbol{n}_{s_2}).$$

When the center of gravities of the plane segments lie on distinct sides of this plane, we distribute the points on either side of the equidistal plane accordingly. Otherwise, we simply associate the points to the closest plane.

## 4 Experiments

We evaluate our approach on Kinect depth images and 3D laser scans as well as on range images from the popular SegComp ABW image dataset [5][3]. The SegComp dataset allows for an objective evaluation of our approach in the context of planar range image segmentation. We compare our approach with results published in [3].

The 30 ABW test images have a resolution of 512×512 pixels. The dataset also provides ground truth segmentation in conjunction with an evaluation tool. Table 1 shows the results of our approach on the SegComp ABW test images at 80% tolerance for the overlap with the ground truth. While our approach is not specifically designed for range

[3] available at http://marathon.csee.usf.edu/seg-comp/SegComp.html

| approach | correct | error | overseg. | underseg. | missed | noise | time (sec) |
|---|---|---|---|---|---|---|---|
| USF | 12.7 (83.5%) | 1.6° | 0.2 | 0.1 | 2.1 | 1.2 | - |
| WSU | 9.7 (63.8%) | 1.6° | 0.5 | 0.2 | 4.5 | 2.2 | - |
| UB | 12.8 (84.2%) | 1.3° | 0.5 | 0.1 | 1.7 | 2.1 | - |
| UE | 13.4 (88.1%) | 1.6° | 0.4 | 0.2 | 1.1 | 0.8 | - |
| OU | 9.8 (64.4%) | - | 0.2 | 0.4 | 4.4 | 3.2 | - |
| PPU | 6.8 (44.7%) | - | 0.1 | 2.1 | 3.4 | 2.0 | - |
| UA | 4.9 (32.2%) | - | 0.3 | 2.2 | 3.6 | 3.2 | - |
| UFPR | 13.0 (85.5%) | 1.5° | 0.5 | 0.1 | 1.6 | 1.4 | - |
| RansacOnly | 6.6 (43.4%) | 2.4° | 1.9 | 0.2 | 6.2 | 7.8 | 13.2 |
| HoughOnly | 4.4 (28.9%) | 3.1° | 0.2 | 0.4 | 9.7 | 3.0 | 3.163 |
| singleRes16 | 5.6 (36.8%) | 1.4° | 0.0 | 0.5 | 8.6 | 1.1 | 0.555 |
| singleRes8 | 10.0 (65.8%) | 1.5° | 0.2 | 0.3 | 4.4 | 1.2 | 0.655 |
| singleRes4 | 7.2 (47.4%) | 1.2° | 1.1 | 0.1 | 6.8 | 7.0 | 1.001 |
| ours | 11.1 (73.0%) | 1.4° | 0.2 | 0.7 | 2.2 | 0.8 | 1.824 |

**Table 1.** Comparison with other segmentation approaches (from [3]) on the SegComp ABW dataset at 80% overlap tolerance. The ground truth images contain $15.2$ regions on average.

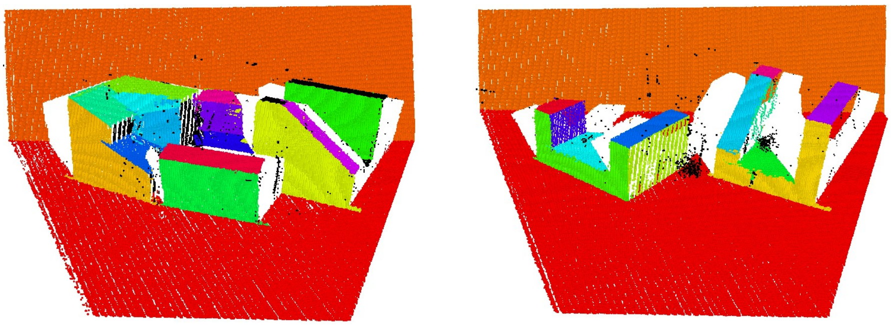

**Fig. 7.** Two segmented scenes from the SegComp ABW dataset.

images, its segmentation quality as well as plane fit accuracy lies in the upper range of results on this dataset. Note, that the best segmentation results have been obtained with methods that exploit the connectivity information encoded in the image structure. This also restricts these methods to the processing of single-view range images. Furthermore, the range images contain strong systematic noise in the form of depth discretization effects, which are difficult to handle for small segments composed of only few points.

In order to assess the contribution of the individual stages of our algorithm, we performed tests with several variants. The method RansacOnly uses a greedy method to detect planes (implemented with the Point Cloud Library, PCL [9]). It iteratively finds the best supported plane fit for the not yet attributed points without using normal information. It only achieves average performance and its run-time strongly depends on the complexity of the scene. HoughOnly is based on our multi-resolution approach but does not perform RANSAC to refine the initial Hough segmentation. It is thus similar to the approach by Vosselman et al. [12]. The HoughOnly method segments the scenes with less accuracy compared to our complete approach. This is attributed to discretiza-

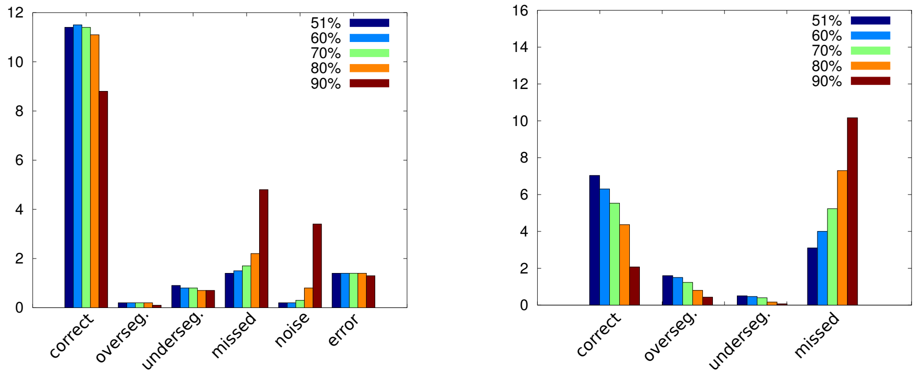

**Fig. 8.** Results of our approach on the SegComp ABW (left) and our Kinect (right) dataset for various overlap tolerances. The ground truth contains $15.2$ and $12.8$ regions on avg., respectively.

tion effects introduced by the accumulator histograms in the Hough transform. In our method, the subsequent RANSAC step filters outliers from the Hough planes and renders the approach robust and accurate. For the methods singleRes16, singleRes8, and singleRes4 we restrict our method to operate on single resolutions (16 cm, 8 cm, 4 cm). The results demonstrate that our multi-resolution approach is important to capture all scales of planes in a scene.

Fig. 8 (left) shows results of our approach on the SegComp ABW test images for different overlap tolerances. It can be seen that the errors in our approach are in large parts due to missed plane segments. As of the high noise, some points may not be assigned to planes or boundaries may not be resolved correctly. Since our method does not consider image neighborhood, it is difficult to achieve $90\%$ overlap. Fig. 7 shows two exemplary segmentations for the ABW dataset. In the left image our algorithm missed multiple planar segments. We attribute some of the misses to the discretization by the octree. This issue could be solved by reprocessing the unsegmented parts in a different discretization.

We also evaluate our approach on depth images of indoor scenes obtained with a Microsoft Kinect camera (s. Fig. 10). Our approach requires ca. 2.06 sec on 640×480 images. In QQVGA resolution (160×120), we are able to process single images in about 106 msec, which allows for real-time applications. We generated a segmentation dataset[4] of 30 images with manually annotated ground truth and evaluate segmentation quality with the SegComp evaluation tool. From Fig. 8 (right) we see that our approach correctly segments $54.9\%$ but only misses $24.2\%$ of the planes at an overlap tolerance of $51\%$. Despite the strong noise of the sensor due to the discretization of the disparity measurements, our approach is able to segment the major planes in the scene. Note that non-planar segments have been annotated as noise in the ground truth. We therefore neglect the noise in the evaluation. Furthermore, for the manual annotation itself it is not possible to achieve perfect overlap with the actual segments in the scene. Since no ground truth of the relative angle between surfaces is available, we do not assess the angular error on this dataset.

[4] available at http://www.ais.uni-bonn.de/download/segmentation/kinect.html

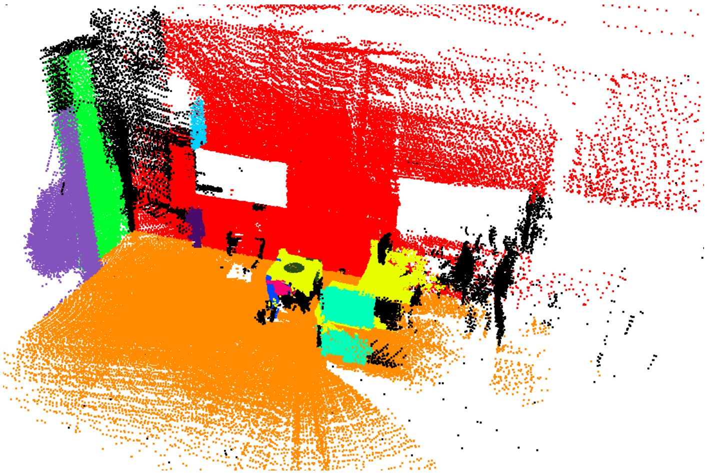

**Fig. 9.** Indoor scene acquired with a 3D laser mounted on a manipulator (black: unsegmented).

In addition to Kinect depth images, we tested our approach on indoor scenes acquired with a laser scanner that is swept with the end-effector of a manipulator (s. Fig. 9). Our approach finds the major plane segments in the building structure. It also finds planar segments in cluttered regions where the sampling density is sufficient.

## 5 Conclusion and Future Work

In this paper we proposed an efficient method for extracting planar segments from 3D point clouds. We combine the Hough transform with robust RANSAC to fit planes on multiple resolutions. By using a coarse-to-fine strategy, we make efficient use of the available data. It allows to consider the largest possible context to make decisions of co-planarity. This also makes our approach data efficient.

In experiments, we demonstrate the robustness and the run-time efficiency of our approach. We compare our method to state-of-the-art approaches using the SegComp database. Our experiments show that we process 3D point clouds of 3D lasers and depth sensors such as the Kinect at high framerates with good quality.

In future work, we will extract further types of geometric shape primitives such as cylinders and spheres. We also plan to tune our approach to the sequential processing of depth images from high framerate sensors such as the Kinect.

## Acknowledgments

This research has been partially funded by the FP7 ICT-2007.2.2 project ECHORD (grant agreement 231143) experiment ActReMa.

## References

1. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. of the ACM (1981)

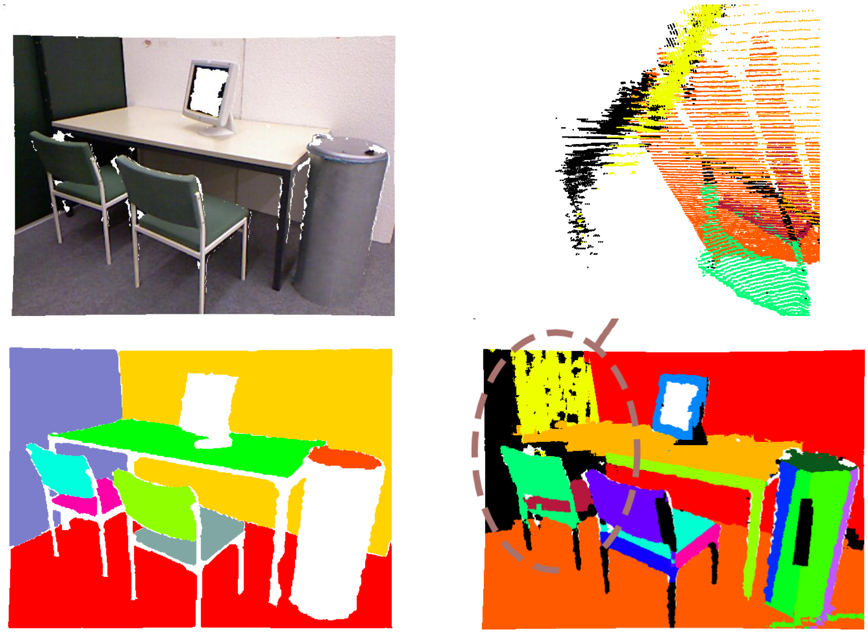

**Fig. 10.** Example Kinect scene (top left), close-view on segmented points in the upper left image corner (top right), ground truth (bottom left), and resulting segmentation (bottom right). White: invalid/unknown, black: unsegmented.

2. Fitzgibbon, A.W., Eggert, D.W., Fisher, R.B.: High-level model acquisition from range images. Computer-Aided Design 29(4), 321 – 330 (1997)
3. Gotardo, P., Bellon, O., Silva, L.: Range image segmentation by surface extraction using an improved robust estimator. In: Proc. of the Int. Conf. on Computer Vision and Pattern Recognition (CVPR) (2003)
4. Harati, A., Gächter, S., Siegwart, R.: Fast range image segmentation for indoor 3D-SLAM. 6th IFAC Symposium on Intelligent Autonomous Vehicles (2007)
5. Hoover, A., Jean-Baptiste, G., Jiang, X., Flynn, P.J., Bunke, H., Goldgof, D.B., Bowyer, K., Eggert, D.W., Fitzgibbon, A., Fisher, R.B.: An experimental comparison of range image segmentation algorithms. IEEE Trans. on Pattern Analysis and Machine Intelligence 18(7), 673 –689 (1996)
6. Hough, P.: Method and means for recognizing complex patterns. U.S. Patent 3.069.654 (1962)
7. Jiang, X., Bunke, H.: Fast segmentation of range images into planar regions by scan line grouping. Machine Vision Applications 7, 115–122 (1994)
8. Rabbani, T.: Automatic reconstruction of industrial installations using point clouds and images. Ph.D. thesis, TU Delft (2006)
9. Rusu, R.B., Cousins, S.: 3D is here: Point cloud library (PCL). In: Proc. of the Int. Conf. on Robotics and Automation (ICRA). Shanghai, China (2011)
10. Schnabel, R., Wahl, R., Klein, R.: Efficient RANSAC for point-cloud shape detection. Computer Graphics Forum 26(2), 214–226 (2007)
11. Taylor, R.W., Savini, M., Reeves, A.P.: Fast segmentation of range imagery into planar regions. Computer Vision, Graphics, and Image Processing 45(1), 42 – 60 (1989)
12. Vosselman, G., Gorte, B.G.H., Sithole, G., Rabbani, T.: Recognising structure in laser scanner point clouds. In: ISPRS - Laser-Scanners for Forest and Landscape Assessment (2004)