Sparse Point Cloud Densification by Using Redundant Semantic Information*

Michael Hödlmoser
CVL, Vienna University of Technology
ken@caa.tuwien.ac.at

Branislav Micusik
AIT Austrian Institute of Technology
branislav.micusik@ait.ac.at

Martin Kampel
CVL, Vienna University of Technology
kampel@caa.tuwien.ac.at

Abstract

This paper presents a novel method for dense 3D reconstruction of man-made environments. Such environments suffer from textureless and non-Lambertian surfaces, where conventional, feature-based 3D reconstruction pipelines fail to obtain good feature matches. To compensate this lack of feature matches, we exploit the semantic information available in 2D images to estimate both a corresponding 3D position and a 3D surface normal for each pixel. A semantic classifier is therefore applied on a single segmented image in order to get a likelihood for a segment providing one of the surface normals within a discrete set of them. To improve the accuracy of this labeling step, we exploit multiple segmentation methods. The global best surface normal configuration over all pixels of an image is then obtained by using a Markov Random Field. In the last step, the 3D model of a single 2D input image is reconstructed by combining the semantic surface normal estimation with the sparse point cloud coming from feature based matching. It is shown experimentally, that our proposed method clearly outperforms state-of-the-art dense 3D reconstruction pipelines and surface layout estimation approaches.

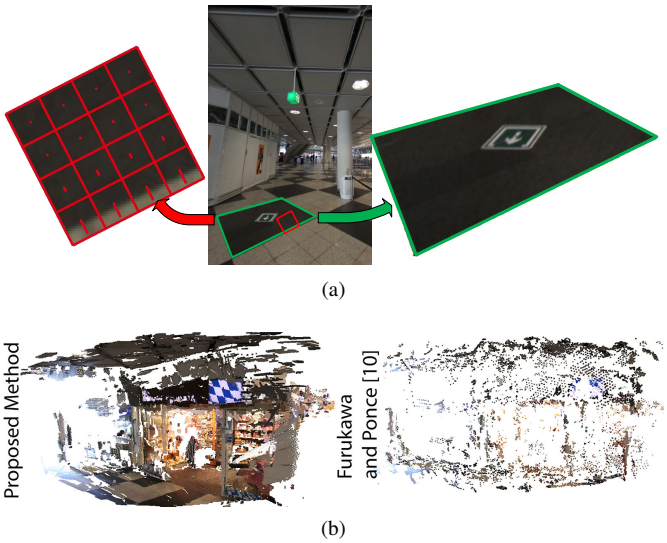


Figure 1: (a) Using semantic meaningful patches (right) increases the likelihood for estimating the 3D surface label correctly compared to using the 2D regions of feature descriptors (*e.g.* SIFT, left). (b) Proposed semantic patch-based 3D reconstruction results (left) compared to conventional feature-based 3D reconstruction results (right) [10].

1. Introduction

Reconstructing a dense 3D model from a single moving camera capturing a real world environment is usually done by generating a sparse point cloud obtained by triangulation [25] followed by a densification [10], where both steps rely on discriminative feature matches. In case of man-made environments, this approach is not feasible because of wrong matches which can occur between corresponding camera views due to similar features obtained from flat and textureless surfaces (*e.g.* walls, floors). Nevertheless, these conventional 3D reconstruction pipelines deliver correct but sparse 3D point clouds where discriminative features can be extracted (*e.g.* posters on the wall, texture on the ground and on the ceiling). As can be seen in Figure 1a, it is hard to say if the 2D region of a feature descriptor (*e.g.* SIFT, see left extracted patch) should be labeled as ground plane / ceiling or as a vertical structure. When analyzing semantic meaningful patches (see right extracted patch) instead of patches coming from feature points the likelihood for obtaining the correct label can be increased.

This paper presents a method which combines a conventional 3D reconstruction pipeline with a patch-based semantic 3D surface normal labeling system in order to overcome the problem of finding discriminative features in man-made environments. As clearly visible in Figure 1b, incorporating patch-based semantic information in our pipeline gives a more planar and complete model compared to exploiting point-based feature matches only, as proposed in [10].

In the first step, we therefore generate a sparse point

*This work was partly supported by FFG FIT-IT projects 835916 (PAMON), 830042 (CAPRI) and CogVis Ltd.

cloud from multiple input images and calculate the 3D camera positions by using the method described in [10]. According to [10], the described method can also generate dense 3D models, which is only true when finding discriminative features. In case of man-made environments, the outcome is also sparse, as can be seen in our experiments section. For the following steps, we operate on a single input image.

We segment the image into semantic meaningful parts using superpixel methods. It is assumed that each segment can be modeled by a planar patch. By using color, texture, perspective features and a boosted decision tree, the 3D plane normal for each segment is estimated by the method of Hoiem *et al*. [17], which in the following is referred to as *semantic labeling*.

In order to be able to perform classification, this normal is chosen from a given set of discrete directions. Superpixel methods are designed to perform well under certain environmental settings (*e.g*. indoor/outdoor setting, specific lighting, defined color, object pose, or different geometric relationships between objects). Since this specific setting cannot cover all possible variations in an image, a superpixel segmentation method may also deliver wrong or missing segments. To compensate these errors, we exploit redundant information and segment the image using multiple superpixel methods [6, 19, 21, 1].

Each pixel is then assigned a possibility to belong to a certain normal orientation class out of the given discrete set. In order to find the global best configuration and therefore the global best normal orientation for each pixel, a Markov Random Field (MRF) is used. By combining the normal estimation with the sparse 3D point cloud, planes are fitted through the 3D points and the cloud is densified. The contribution of this work is therefore two-fold: first, semantic information is used to compensate missing and wrong feature matches at textureless and non-Lambertian surfaces in man-made environments. Second, redundant information in terms of multiple segmentation methods is used to exploit the advantages of each single one to obtain a higher accuracy for both the surface labeling and the 3D reconstruction.

2. Related Work

For the last few years, transferring this human ability to computers is one of the grand challenges in computer vision. Having the 3D geometry of a scene would help applications placed on top of this knowledge. Assuming a given ground plane is for example necessary for initializing the tracking sequence, but it also helps in tasks such as autonomous robot navigation and automatic object manipulation. As our implementation is a combination of single view reconstruction and 3D reconstruction using multiple images, we present related work for both areas in the following.

2.1. Single View 3D Reconstruction

Reconstructing 3D models from a single 2D image is an ill-posed problem. Nevertheless, the related work presented in the following segments an image into geometrically meaningful classes in order to enable the 3D reconstruction from a single image. Each pixel is then assigned a geometric label. The method of [16] automatically constructs a rough 3D model from a single 2D image. This is established by learning a statistical model of surface normal label classes. Extracting 3D information from a single 2D image showing a Manhattan world indoor environment is described in [5]. They assume to have a calibrated camera, extract edges, the ground plane and surface orientations from the images and obtain a final labeling by solving an MRF. Labeling the 3D layout of a scene was published in [17]. By combining multiple 2D cues (color, texture and perspective features of a patch) the classifier is trained on multiple indoor and outdoor still images using boosted decision trees. Each image is segmented using the approach presented in [6]. For getting a higher accuracy, the method merges segments to obtain different segments in terms of size and shape. Gould *et al*. [12] obtained a holistic representation of the scene by finding semantic and geometric meaningful and consistent regions in the image. Following [18], the layout of indoor Manhattan World scenes can be estimated from a single image. By connecting and sweeping line segments, the most likely box layout is found. Hedau *et al*. [15] presented a novel approach on estimating the scene layout of cluttered rooms by fitting the most likely 3D box. Another segmentation and depth estimation framework using an MRF and semantic segmentation using meanshift was presented in [20]. Gupta *et al*. [13] presented an approach which allows estimating the 3D scene layout by combining volumetric reasoning (*e.g*. occlusions, arrangement of objects) with reasoning with mechanics (*e.g*. material density and internal energy). Schwing *et al*. [24] proposed to estimate the 3D surface layout of an indoor scene by decomposing higher order potentials into pairwise potentials by incorporating integral images to geometry. Bedroom sampling on still images by incorporating the geometric features of objects within a room is used to obtain a rough layout of the room in [23].

2.2. Multiple View 3D Reconstruction

Having multiple images helps to generate a sparse 3D point cloud by using Structure from Motion (SfM). A sparse 3D reconstruction framework was introduced in [25]. The algorithm creates a sparse point cloud in combination with corresponding camera positions from a given image set, where the reconstruction is done incrementally. A dense 3D reconstruction pipeline was published in [10]. By using multiple features and multiple iterations of matching, expanding and filtering these matches, a dense model is

obtained from multiple images. Dense reconstruction of well-known touristic parts of cities is presented in [2] by using a parallel distributed system. Dense reconstruction processed on a single computer is presented by [9]. Images from tourists are collected and matched by the method published by [2] to obtain the 3D model. Automatic dense 3D reconstruction from 2D images using planar patches to recover both planar and non-planar structures was introduced by [11]. Planes are detected using RANSAC and automatically linked for multiple view reconstruction. Xiao and Furukawa [27] presented an algorithm for reconstruction and visualization of large scale indoor environments from various museums. By exploiting volumetric primitives and therefore doing a volumetric reconstruction instead of recovering a surface model, wall configurations are found and textured. Häne *et al*. [14] presented a pipeline for piecewise planar depth map fusion and 3D reconstruction using a first-order primal dual optimization method instead of a higher order one.

Similar to our approach, sparse 3D point clouds can also be combined with information coming from single images. Brostow *et al*. [4] published an approach for labeling 2D video sequences from outdoor scenes using sparse 3D point clouds. By using Delaunay Triangulation, a relief mesh is set up from the 3D points. Based on the orientation and location of the triangles, the traffic scene is segmented. For this approach, the features are purely calculated on the geometric observations. A 3D reconstruction pipeline using a single semantic segmentation and matching method was presented in [22]. Flint *et al*. [7] presented a method which incorporates stereo, monocular and 3D features to iteratively help in the segmentation process of indoor videos. Bayesian filtering with motion cues of possible hypotheses of box layouts of input videos showing indoor scenes is presented in [26]. An approach using videos of street scenes for segmenting the scene was presented in [28]. Reconstructed 3D points are manually but not perfectly labeled to help in the 2D segmentation process which is performed using an MRF. Floros *et al*. [8] presented an approach which combines spatial and temporal smoothness terms between corresponding pixels in a single higher-order CRF in order to obtain an image segmentation formulation.

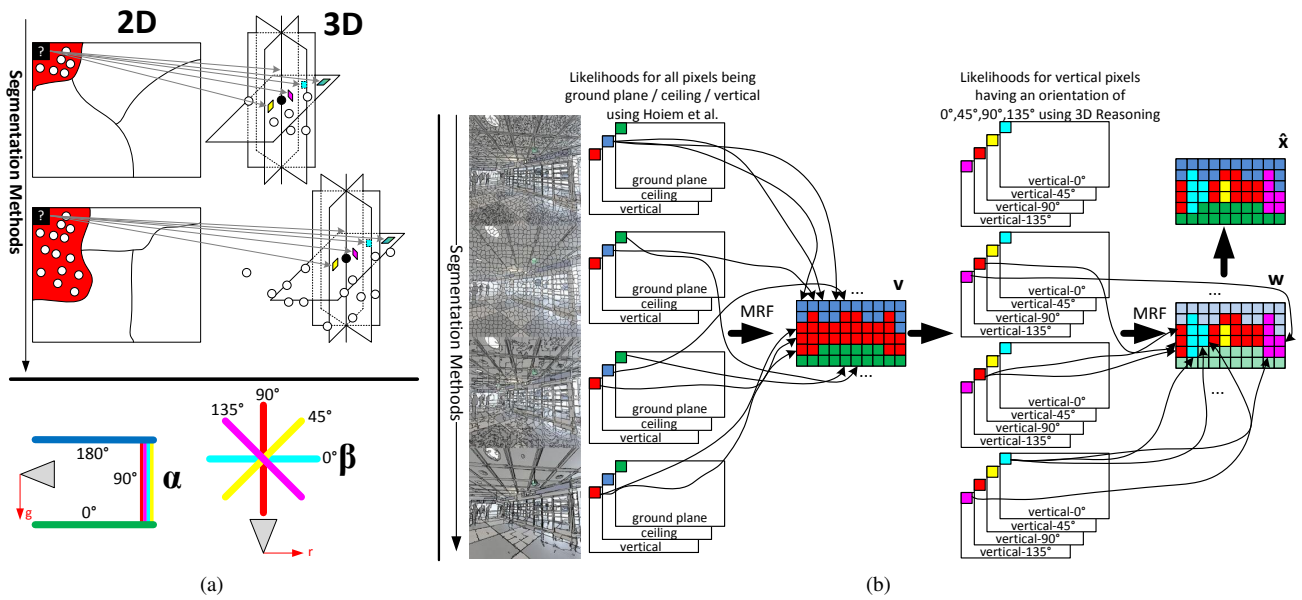
This paper deals with the problem of wrong or missing feature matches between corresponding views by also incorporating semantic information. Different to existing approaches, the proposed approach tackles the specificity of a single algorithm and increasing its robustness by exploiting redundancy. Redundant information for getting corresponding 3D points for each 2D pixel is achieved by combining different segmentation methods when performing semantic reasoning.

3. 3D Point Cloud Densification

The goal of our pipeline is to do both, classifying each pixel according to the labels *ground plane*, *ceiling* and *vertical*, defined by an angle α and *vertical-*β (where $\alpha = 90°$ corresponds to all possible vertical orientations β) and gathering a 3D point for each 2D pixel by only considering a single input image and a sparse point cloud of the scene (see bottom image of Figure 2a). The angle α can take values of a discrete set $\mathcal{L}_1 = \{0°, 90°, 180°\}$ and describes the orientation difference between the camera's gravity vector $\mathbf{g}$ and the plane normal $\mathbf{n}$. The orientations $\alpha = 0°$ can be seen as the *ground plane* of the scene, $\alpha = 90°$ can be seen as any *vertical structure*, $\alpha = 180°$ corresponds to the *ceiling*. The angle β can take values from a discrete set $\mathcal{L}_2 = \{0°, 45°, 90°, 135°\}$ and describes the orientation difference between the camera's right vector $\mathbf{r}$ and the plane normal $\mathbf{n}$.

The workflow of the proposed algorithm can be seen in Figure 2. In the first step, a sparse 3D point cloud and corresponding camera positions are generated from multiple input images by using the method described in [10]. As can be seen in the top image of Figure 2a, the pixel in question is shown as rectangle in the image for demonstration purposes. All the pixels within the segment which also holds the pixel in question are then projected on all the planes in question having orientation α and β. The 3D center point of the planes (denoted as black circle in the image) is obtained by calculating the median 3D point from those points from the 3D point cloud which are located within the segment's 2D area when projected onto the image plane. This means that multiple corresponding 3D points are available for each 2D pixel. Each normal direction within a discrete set of normals is assigned a certain likelihood coming from semantic and 3D reasoning, as can be seen in Figure 2b. The globally best result and therefore the best fitting corresponding 3D point is obtained by pixelwise optimization using an MRF. As the likelihoods from [17] cannot be compared with likelihoods from 3D reasoning, the optimization is done in two steps to obtain the optimized results $\mathbf{v}$, $\mathbf{w}$ and their combination $\hat{\mathbf{x}}$.

After generating the point cloud from an image sequence, the algorithm is operating on a single image, which is splitted into semantically meaningful parts. We assume that each segment in the image can be represented by a planar patch. The main problem in this step is that segmentation methods are designed for specific environmental settings (*e.g*. certain lighting conditions, specific objects or scenes *etc*.), which means that they may provide wrong or missing segments when these settings or conditions are not met. To exploit the advantages from several segmentation methods in order to increase the accuracy of the 3D reconstruction pipeline, we segment each frame by using multiple segmentation methods. The outcome of the su-



(a) (b)

Figure 2: (a) Top: Multiple segmentation methods generate multiple 3D points for each pixel. Bottom: Each pixel can be labeled with one of the labels within the discrete set of orientations. (b) Workflow of the proposed optimization.

perpixel algorithms described in [6, 19, 21, 1] is shown in Figure 3 from left to right. As can be seen, there is a variation between shape and size of the segments, which we want to exploit for both labeling and 3D reconstruction. To

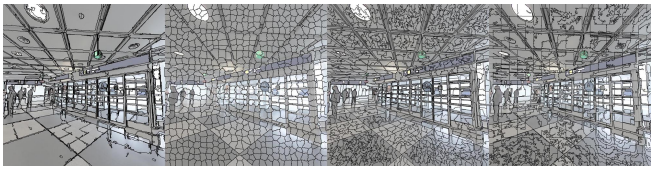


Figure 3: Multiple superpixel segmentation methods deliver different results in terms of shape and size of the segments. From left to right: [6], [19], [21], [1].

get a spatial consistent result for the whole image, the label for each pixel is determined independently of the segments of an image. The segmentation methods therefore serve as soft priors to the final labeling problem. The solution to this problem corresponds to finding the configuration of a Gibbs distribution with maximal probability, which is equivalent to finding the maximum posterior (MAP) configuration of an MRF. Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a graph described by vertices $\mathcal{V}$, which in this case are represented by the pixels of the image, and edges $\mathcal{E}$. When having a set of random variables $\mathbf{X} = \{X_1, X_2, \dots X_N\}$ and a label configuration $\mathbf{x} = \{x_1, x_2, \dots x_N\}$ which can take values from the discrete set of labels $\mathcal{L}_1$, the energy term E of the pairwise MRF is defined by

$$E(\mathbf{x}) = \sum_{i \in \mathcal{V}} \psi_i(x_i) + \sum_{i \in \mathcal{V}, j \in \mathcal{N}_i} \psi_{i,j}(x_i, x_j), \qquad (1)$$

where $\mathcal{N}_i$ is the neighborhood of node i, ψ_i is the unary potential in the graph and $\psi_{i,j}$ is the pairwise potential, or smoothness term, between neighboring pixels. The following part demonstrates how to find the unary and binary potentials.

The method of Hoiem *et al.* [17] then delivers a likelihood for each segment having an orientation α. The likelihood u for a single pixel y_i within the segment s at image location i is given by

$$u(y_i) = P(\alpha | y_i). \qquad (2)$$

As the original implementation differentiates between left and right wall segment, we combine these two likelihoods to obtain the likelihood for the label *vertical* ($\alpha = 90°$). We improve the results obtained from [17] by also calculating the horizon line from vanishing points, which are extracted from the image. These points are obtained by detecting lines and clustering them by using RANSAC. This prevents from labeling pixels above the horizon line as *ground plane* and pixels below the horizon line as *ceiling*. The unary potentials and smoothness terms are set to

$$\psi_i(x_i) = exp(-u(y_i)) \cdot \lambda \cdot d_s \qquad (3)$$

$$\psi_{i,j}(x_i, x_j) = 1 - exp\left(-\left|\frac{(\alpha(y_i) - \alpha(y_j))}{180}\right|\right) \qquad (4)$$

where $u(y_i)$ is obtained by using Hoiem's method, d_s is the 3D Euclidean distance between the center point of segment s (which contains y_i) and the camera center, λ is a normalizing constant and $\alpha(y_i)$ is the surface normal orientation at pixel y_i. Normalization is reached by dividing through 180. The distance d_s is used to increase the likelihood for pixels which are closer to the camera center. Note that both $u(y_i)$ and d_s are obtained for each discrete surface normal orientation and for each segmentation result. When using 4 different segmentation methods, this leads to $4 \cdot |\mathcal{L}_1| = 12$ labels a pixel can obtain. The MAP configuration $\mathbf{v} = \{v_1, v_2, \dots v_N\}$ is then found by

$$\mathbf{v} = \underset{\mathbf{x}}{\operatorname{argmin}}\, E(\mathbf{x}). \tag{5}$$

In our implementation we choose an 8-connectivity so that each pixel has eight neighboring pixels. The MRF is then solved by using Iterated Conditional Modes (ICM) [3].

After having defined if the pixel is located on the ground, the ceiling, or a wall segment, we want to find out the orientation of the wall. This step is referred to as *3D reasoning*. Since unary potentials between Hoiem's approach and 3D reasoning cannot be directly compared, we solve a second pairwise MRF between neighboring pixels to find this orientation. For finding the unary potential, all projected 3D points which are within the area of a segment are investigated for each segment. If a certain number of points (5 in our experiments) is found within the 2D area of a segment, we fit a plane through all these 3D points by exploiting RANSAC. The 3D center point of the segment is obtained by gathering the median of all points marked as inlier in the plane fitting step. In case there are not enough points, a segments normal is set as the median from its 3 neighboring segments' ones. We then calculate the angle $\phi(\beta)$ between the normal $\mathbf{n}$ of the fitted plane and the normal $\mathbf{n}(\beta)$ of the plane in question by

$$\begin{aligned} d(\beta) &= \frac{\cos^{-1}\left(\mathbf{n} \cdot \mathbf{n}(\beta)\right)}{\|\mathbf{n}\|\|\mathbf{n}(\beta)\|} \\ \phi(\beta) &= \min(d(\beta), \pi - d(\beta)). \end{aligned} \tag{6}$$

The label configuration $\mathbf{x}$ can take values from the discrete set of labels $\mathcal{L}_2$. Unary and pairwise terms for the second MRF are set to

$$\psi_i(x_i) = 1 - exp\left(-\frac{\phi(\beta)}{180}\right) \tag{7}$$

$$\psi_{i,j}(x_i, x_j) = 1 - exp\left(-\left|\frac{(\beta(y_i) - \beta(y_j))}{180}\right|\right) \tag{8}$$

where $\beta(y_i)$ is the surface normal orientation for pixel y_i. Once more, when using 4 different segmentation methods, this leads to $4 \cdot |\mathcal{L}_2| = 16$ different labels a pixel can obtain. The final configuration $\mathbf{w} = \{w_1, w_2, \dots w_N\}$ is then found similarly to solving Equation 5.

	Hoiem[17]	Proposed (SM)	Proposed (MM)
ground plane	68.39	85.54	87.41
ceiling	43.19	75.44	79.61
vertical	22.11	36.39	39.94
global	60.17	82.64	85.19
average	44.56	65.97	68.99

Table 1: Percentage of correctly classified pixels per label for the *Airport* dataset.

At this stage, we know which pixel needs to be classified as either *ground plane*, *ceiling*, or *vertical*. We also know which orientation is the globally best one when a pixel is classified as *vertical*. In the last step, $\mathbf{v}$ and $\mathbf{w}$ are combined to obtain the final labeling $\hat{\mathbf{x}}$ by

$$\hat{\mathbf{x}} = \begin{cases} v_i & \text{if } \alpha(y_i) \in \{0°, 180°\} \\ w_i & \text{else} \end{cases} \tag{9}$$

For both classification steps, not only a 2D segmentation of the scene is obtained but also corresponding 3D points for each 2D pixel. As stated previously, the 3D center point of each segment is obtained and a plane, where the orientation corresponds to the 2D orientation label obtained in the optimization step, is fitted through it. As the orientation is known at this stage, it is also known from which 2D segmentation the best configuration comes from. For each 2D pixel, the 3D point corresponding to the globally best segmentation result is therefore used.

4. Experiments

Experiments are conducted using an indoor dataset holding 270 images showing the indoor environment of an Airport having multiple height layers and complex geometric structures. The images are taken by a person walking on the ground plane with a Canon EOS 5D Mark II. A sparse point cloud and the 3D camera positions are obtained by using all the images in the dataset and the method of Furukawa, described in [10]. Segmentation is done by using the algorithms described in [6, 19, 21, 1]. For visualization purposes, all 3D points which are too far away from the camera image plane are sorted out. Therefore, an empirically found threshold of ten times the distance between the current and the subsequent camera center is chosen.

4.1. Quantitative Experiments

To show the relevance of the MRF in this approach, the 2D labeling results of Hoiem's method are compared to our proposed method. This means that the label configuration can only take values of $\alpha \in \mathcal{L}_1$. We manually labeled 100 ground truth images where pixels which are left blank in the ground truth image are not taken into account for the

comparison. Note that these pixels are marked in black in Figure 4a. Having a ground truth labeled image $\mathcal{G}$ and a resulting image $\mathcal{R}$, the accuracy p_l for label l is determined by

$$p_l = \frac{|\mathcal{G}_l \cap \mathcal{R}_l|}{|\mathcal{G}_l \cup \mathcal{R}_l|}, \tag{10}$$

where $|\cdot|$ refers to the number of pixels assigned to a certain discrete angle α. The percentage of correctly classified pixels for each label can be seen in Table 1, Figure 4a shows some sample results for the basic approach of Hoiem *et al*., the proposed MRF approach using a single segmentation method (SM) [6] and the proposed MRF approach using four segmentation methods (MM). The ground plane is labeled in green, the ceiling in blue and vertical segments in red. As can be seen, using multiple segmentation methods increases the accuracy of labeling the pixel correctly.

4.2. Qualitative Experiments

After having the labeled images, the reconstruction is established by using the method described in Section 3. Figure 4b shows two sample 3D model results rendered from novel viewpoints. As can be seen, planar patches are reconstructed densely where segments closer to the camera center are reconstructed denser than those patches farther away. As demonstrated by the quantitative experiments, using multiple segmentation methods improves the 2D labeling accuracy. Figure 5 shows a qualitative comparison between 3D reconstruction results using a single superpixel method (SM) [6] and multiple ones (MM) [6, 19, 21, 1]. As can be seen, using multiple segmentation methods improves the results of both the 2D labeling as well as the 3D reconstruction. It is also clearly visible in the 2D labeling images that more errors occur at patches which are farther away from the camera center. As the goal of this method is to densify a sparse point cloud, Figure 6 shows several sample results, where each row shows the reconstruction results for one image. Each column presents the input image, the reconstructed model using the proposed method and the resulting 3D point cloud obtained when using Furukawa's method [10]. Please see our supplementary material for two sample 3D models obtained by using our method and the method proposed by [10].

5. Conclusion

We presented a novel 3D reconstruction pipeline which performs densification of sparse point clouds obtained from man-made environments. Conventional feature-based 3D reconstruction pipelines may deliver wrong and missing matches due to specular and textureless surfaces in these environments. This work tries to overcome this problem by combining features with semantic 2D information. A sparse point cloud and corresponding 3D camera positions are therefore obtained from conventional methods using multiple input images. A single image is then segmented and a likelihood for each segment having a certain 3D surface orientation is calculated. Multiple segmentation methods are used to exploit the advantages of each single one in order to obtain a higher accuracy for both this labeling step and the subsequent reconstruction step, which is performed in combination with the sparse point cloud. As can be seen in the experiments section, the proposed method achieves denser and more accurate results compared to state-of-the-art labeling and 3D reconstruction pipelines.

References

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Suesstrunk. SLIC Superpixels Compared to State-of-the-art Superpixel Methods. *PAMI*, 2012. 2, 4, 5, 6

[2] S. Agarwal, N. Snavely, I. Simon, S. M. Seitz, and R. Szeliski. Building rome in a day. In *ICCV*, pages 72–79, 2009. 3

[3] J. Besag. On the statistical analysis of dirty pictures. *Journal of the Royal Statistical Society, Series B (Methodological)*, 48(3):259–302, 1986. 5

[4] G. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla. Segmentation and recognition using structure from motion point clouds. In *ECCV*, pages 44–57, 2008. 3

[5] E. Delage, H. Lee, and A. Ng. A dynamic bayesian network model for autonomous 3d reconstruction from a single indoor image. In *CVPR*, pages 2418–2428, 2006. 2

[6] P. Felzenszwalb and D. Huttenlocher. Efficient graph-based image segmentation. *IJCV*, 59(2):167–181, 2004. 2, 4, 5, 6

[7] A. Flint, D. Murray, and I. Reid. Manhattan scene understanding using monocular, stereo, and 3d features. In *ICCV*, 2011. 3

[8] G. Floros and B. Leibe. Joint 2d-3d temporally consistent semantic segmentation of street scenes. In *CVPR*, pages 2823–2830, 2012. 3

[9] J.-M. Frahm, P. F. Georgel, D. Gallup, T. Johnson, R. Raguram, C. Wu, Y.-H. Jen, E. Dunn, B. Clipp, and S. Lazebnik. Building rome on a cloudless day. In *ECCV*, pages 368–381, 2010. 3

[10] Y. Furukawa and J. Ponce. Accurate, dense, and robust multi-view stereopsis. In *CVPR*, 2007. 1, 2, 3, 5, 6, 8

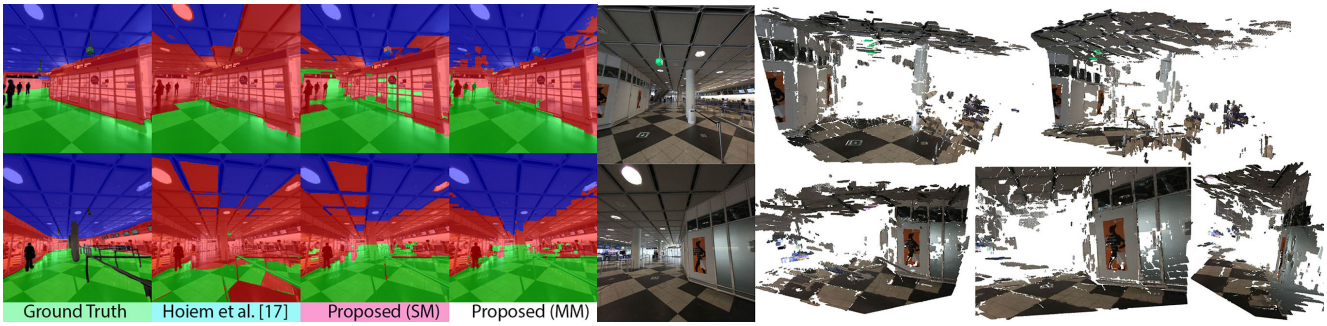
[11] D. Gallup, J.-M. Frahm, and M. Pollefeys. Piecewise planar and non-planar stereo for urban scene reconstruction. In *CVPR*, pages 1418–1425, 2010. 3

[12] S. Gould, R. Fulton, and D. Koller. Decomposing a scene into geometric and semantically consistent regions. In *ICCV*, pages 1–8, 2009. 2

[13] A. Gupta, A. Efros, and M. Hebert. Blocks world revisited: Image understanding using qualitative geometry and mechanics. In *ECCV*, pages 482–496, 2010. 2

[14] C. Häne, C. Zach, B. Zeisl, and M. Pollefeys. A patch prior for dense 3d reconstruction in man-made environments. In *3DIMPVT*, pages 563–570, 2012. 3

[15] V. Hedau, D. Hoiem, and D. Forsyth. Recovering the spatial layout of cluttered rooms. In *ICCV*, 2009. 2



(a) (b)

Figure 4: (a) Sample labeling results where ground plane = green, ceiling = blue, vertical = red. (b) 3D models rendered from novel viewpoints using the results obtained by the proposed method.

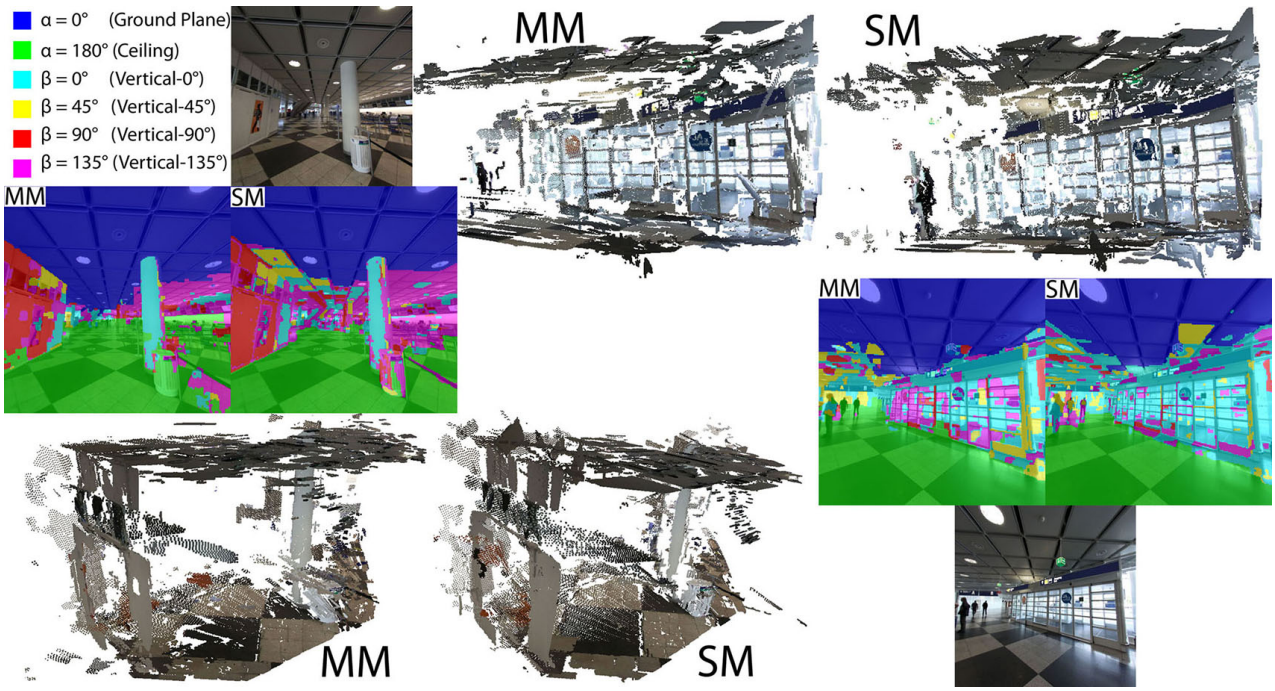


Figure 5: Qualitative comparison between 2D labeling and 3D models obtained by using a single segmentation method (SM) and four different ones (MM).

[16] D. Hoiem, A. Efros, and M. Hebert. Automatic photo pop-up. *ACM Transactions on Graphics*, 24(3):577–584, 2005. 2

[17] D. Hoiem, A. A. Efros, and M. Hebert. Recovering surface layout from an image. *IJCV*, 75(1), oct 2007. 2, 3, 4, 5

[18] D. Lee, M. Hebert, and T. Kanade. Geometric reasoning for single image structure recovery. In *CVPR*, 2009. 2

[19] A. Levinshtein, A. Stere, K. Kutulakos, D. Fleet, S. Dickinson, and K. Siddiqi. Turbopixels: Fast superpixels using geometric flows. *PAMI*, 31(12):2290–2297, 2009. 2, 4, 5, 6

[20] B. Liu, S. Gould, and D. Koller. Single image depth estimation from predicted semantic labels. In *CVPR*, 2010. 2

[21] M.-Y. Liu, O. Tuzel, S. Ramalingam, and R. Chellappa. Entropy rate superpixel segmentation. In *CVPR*, pages 2097–2104, 2011. 2, 4, 5, 6

[22] B. Micusik and J. Kosecka. Piecewise planar city 3d modeling from street view panoramic sequences. In *CVPR*, pages 2906–2912, 2009. 3

[23] L. D. Pero, J. Bowdish, D. Fried, B. Kermgard, E. Hartley, and K. Barnard. Bayesian geometric modeling of indoor scenes. In *CVPR*, 2012. 2

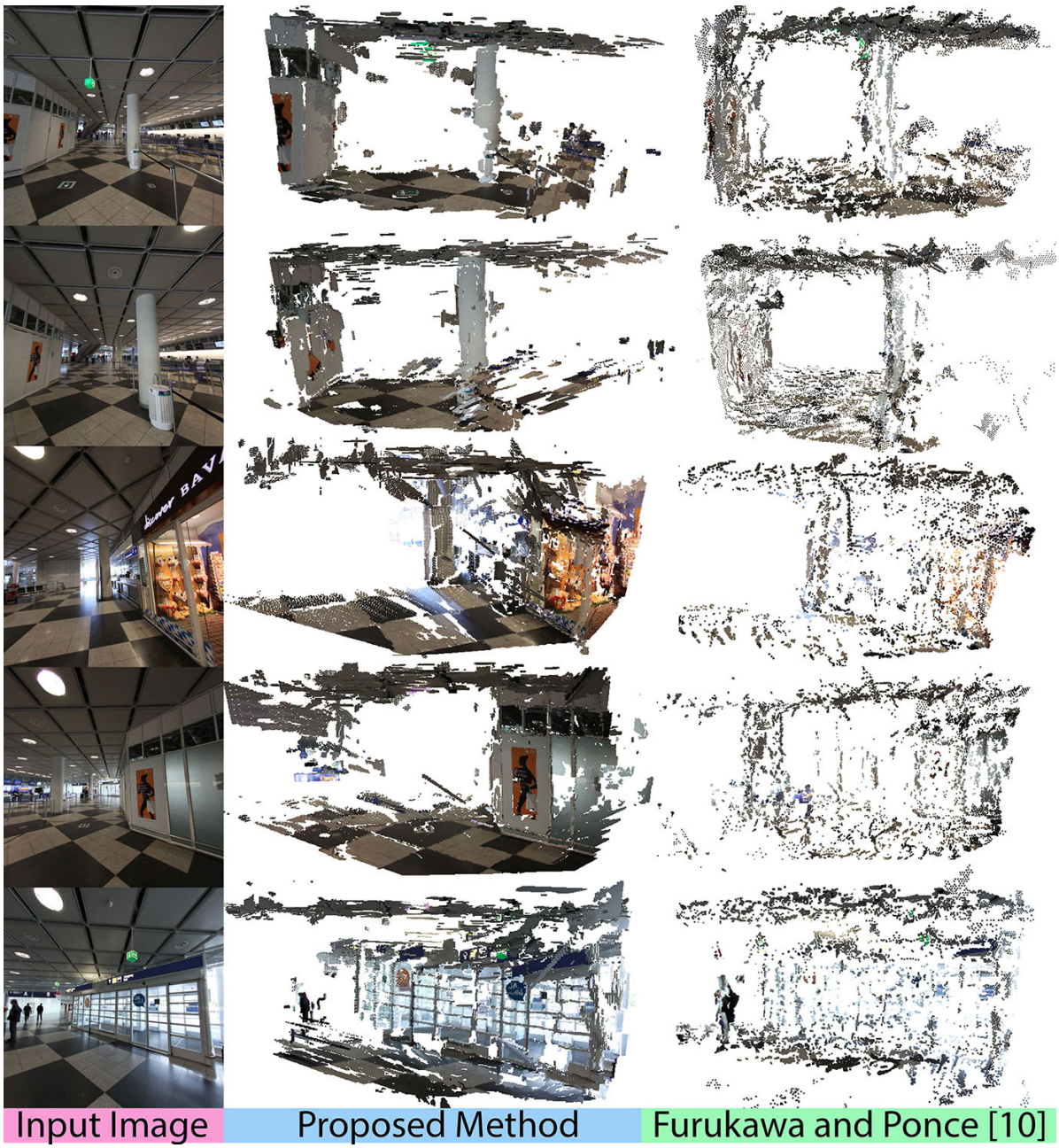


Figure 6: Qualitative results. Each row presents results using the input image shown in column 1. The following columns show the reconstructed scene using the proposed method and Furukawa's method [10]. This figure is best viewed in color. See text for details.

[24] A. Schwing, T. Hazan, M. Pollefeys, and R. Urtasun. Efficient structured prediction for 3d indoor scene understanding. In *CVPR*, pages 2815–2822, 2012. 2

[25] N. Snavely, S. M. Seitz, and R. Szeliski. Photo tourism: Exploring photo collections in 3d. In *ACM Transactions on Graphics*, pages 835–846, 2006. 1, 2

[26] G. Tsai, C. Xu, J. Liu, and B. Kuipers. Real-time indoor scene understanding using bayesian filtering with motion cues. In *ICCV*, pages 121–128, 2011. 3

[27] J. Xiao and Y. Furukawa. Reconstructing the world's museums. In *ECCV*, pages 668–681, 2012. 3

[28] J. Xiao and L. Quan. Multiple view semantic segmentation for street view images. In *ICCV*, pages 686–693, 2009. 3