**2018 IEEE Intelligent Vehicles Symposium (IV)**
**Changshu, Suzhou, China, June 26-30, 2018**

# Planecell: Representing Structural Space with Plane Elements

Lei Fan[1,2], Long Chen[2], Kai Huang[2] and Dongpu Cao[3]

*Abstract*— **Reconstruction based on the stereo camera has received considerable attention recently, but two particular challenges still remain. The first concerns the need to present and compress data in an effective way, and the second is to maintain as much of the available information as possible while ensuring sufficient accuracy. To overcome these issues, we propose a new 3D representation method, namely, planecell, that extracts planarity from the depth-assisted image segmentation and then directly projects these depth planes into the 3D world. The proposed method demonstrates its advancement especially dealing with large-scale structural environment, such as autonomous driving scene. The reconstruction result of our method achieves equal accuracy compared to dense point clouds and compresses the output file 200 times. To further obtain global surfaces, an energy function formulated from Conditional Random Field that generalizes the planar relationships is maximized. We evaluate our method with reconstruction baselines on the KITTI outdoor scene dataset, and the results indicate the superiorities compared to other 3D space representation methods in accuracy, memory requirements and the scope of applications.**

## I. INTRODUCTION

3D reconstruction has been an active research area in the computer vision community, which can be used in numerous tasks, such as perception and navigation of intelligent robotics, high precision mapping, and online modeling. Among various sensors that can be used for reconstruction, stereos cameras are popular for offering advantages in terms of being low-cost and supplying color information. Many researchers have improved the precision and speed of self-positioning and depth calculation algorithms to enable better reconstruction. However, the basic map representation method determines the upper bound of reconstruction performance to some extent. Current approaches including point clouds, voxel-based or piece-wise planar methods are confronted with problems dealing massive stereo image sequences, such as significant redundancy, ambiguities and high memory requirements. To overcome these limitations, we propose a new representation method named *planecell*, which models planes to deliver geometric information in the 3D space.

It is a classical approach to representing the 3D space with a preliminary point-level map. The point-based representations usually suffer a tradeoff of density and efficiency. Many approaches [17], [1], [14] have been developed to address this issue, i.e., to merge similar points in the 3D reconstruction results for both indoor and outdoor scenes. The current leading representation method, called the voxel map [2], [22], [17], [20], is designed to give each voxel grid an occupancy probability, and then aggregates all points within a fixed range. However, dense reconstructions using regular voxel grids are limited to reach small volumes because of their memory requirements.

Previous studies have adopted the plane prior both in stereo matching [23] and reconstruction [17], [18], [10], [3]. Deriving primitives in the model raises the complexity, which restricts further applications. The structure-from-motion method [3] presented the urban scene with planes underlying sparse data. Superpixels or image segmentation methods have been applied in the representation [4] as basic components. Combined with meshes and smoothing terms, it achieves good results on large-scale scenes. Although these methods can reconstruct the scene in a dense and lightweight approach, the accuracy and time-consumption are still unsatisfactory.

In this paper, we propose a novel approach that differs by mapping the 3D space with basic plane units directly extracting from 2D images, which is called planecell for it resembles cells to a living being. The proposed method utilizes a general function to represent a group of points with similar geometric information, i.e., belong to the same plane by a depth-aware superpixel segmentation, and these planes are projected into the real-world coordinates after plane-fitting with depth values. The standardized representation promotes memory efficiency and provides convenience for following computations, such as large surface segmentation and distance calculation. Our method extracts planecells from images by superpixelizing the input image following the hierarchical strategy of SEEDS [19] and converts them into a 3D map. Further aggregation of planecells to a larger surface is modeled by a Conditional Random Field (CRF) formulation. The proposed representation is motivated by the planar nature of the environment. The input to our method is stereo pairs, and the output is a plane-based 3D map with decent pixel-wise precision and high compression rate. Note the depth acquirement is not specified, e.g. stereo matching algorithms, LiDAR or RGB-D sensors can also be used.

The detailed contributions of this paper are as follows: (a) We propose a novel plane-based 3D map representation method that demonstrates remarkable accuracy and has enhanced the space perception abilities. (b) The proposed method reduces the required memory of presenting large-

This work was supported in part by the National Natural Science Foundation of China under Grant 61773414.

[1]Lei Fan is with the Vehicle Intelligence Pioneers Inc., Qingdao Shandong 266109, P.R.China chenl46@mail.sysu.edu.cn

[2]Lei Fan, Long Chen and Huang kai are with School of Data and Computer Science, Sun Yat-sen University, Guangzhou, Guangdong, P.R.China. chenl46@mail.sysu.edu.cn

[3]Dongpu Cao is with Department of Mechanical and Mechatronics Engineering, University of Waterloo, 200 university avenue west, Waterloo, Ontario N2L3G1, Canada. dongpu@uwaterloo.ca

978-1-5386-4451-5/18/$31.00 ©2018 IEEE

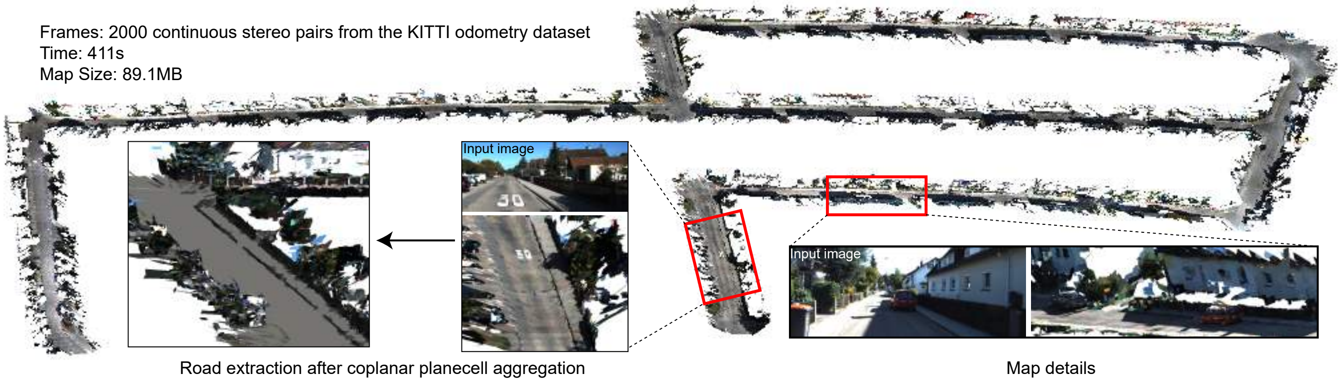

Frames: 2000 continuous stereo pairs from the KITTI odometry dataset
Time: 411s
Map Size: 89.1MB
Input image
Input image
Road extraction after coplanar planecell aggregation
Map details

Fig. 1. The map reconstructed by proposed method. The result shows three advantages: (a) the ability of dealing with large-scale reconstruction; (b) the low-loss of accuracy (detailed quantitative evaluation is processed on KITTI stereo dataset compared to ground truth); (c) the low requirements of both time and memory.

scale maps with innovative superpixel segmentation. (c) The expansibility and efficiency of our representation method are studied in applications include but not limited to road extraction and obstacle avoidance in the experiment. In practice, this objective can be optimized on a single core in as little as $0.2$ seconds for about 700 planecells. To further aggregate coplanar planes requires only $0.1$s per frame. The remainder of this paper is organized as follows: Section II briefly reviews the related literature. The overview of the proposed method is introduced in Section III. The detail of our method is explained in Section IV. In Section V, we demonstrate the 3D representation. The experimental results are conducted in Section VI. In Section VII, we conclude this paper.

## II. RELATED WORK

Basic 3D map representation methods using an image pair are inheritors of various stereo matching algorithms [23], [15], [24]. Point-based 3D reconstruction methods directly transforming stereo matching results lack structural representations. Recent point-level online scanning [25] produces a high-quality 3D model of small objects with the geometric surface prior, which is simpler to operate than strong shape assumptions. For large-scale reconstructions, sparse point-based representations are mainly used for their quality and speed. The point-based maps embedded in the system [8] is designed for real-time applications, such as localization. Adopting denser point clouds in the mapping is challenging because it involves managing millions of discrete values.

The heightmap is a representation adopting 2.5D continuous surface representations, which shows its advancement modeling large buildings and floors. Gallup et al. proposed an $n$-layer heightmap [11] to support more complex 3D reconstruction of urban scenes. The proposed heightmap enforced vertical surfaces and avoided major limitations when reconstructing overhanging structures. The basic unit of heightmap is the probability occupancy grid computed by the bayesian inference, which could compress surface data efficiently but is also lossy of point-level precision.

Recent studies on voxelized 3D reconstruction focus on infusing primitives into the reconstructions [16], [17], [9], [5] or utilizing scalable data structures to meet CPU requirements [20]. Dame et al. proposed a formulation [6] which combines shape priors-based tracking and reconstruction. The map was represented as voxels with two parameters including the distance to the closest surface and the confidence value. Nonetheless, the accuracy of volumetric reconstruction is always limited to itself, and re-estimating object surfaces from voxels or 3D grids leads to ambiguities.

Planar nature assumptions have been applied to both the reconstruction [10], [4] and depth recovery from a stereo pair [23], [21]. For surface reconstruction or segmentation, Liu et al. [16] partitioned the large-scale environment into structural surfaces including planes, cylinders, and spheres using a higher-order CRF. A bottom-up progressive approach is adopted alternately on the input mesh with high geometrical and topological noises. Adopting this assumption, we present a new representation method of 3D space, which is composed of planes with pixel-level accuracy.

## III. SYSTEM OVERVIEW

As shown in Fig. 2, the input to the system is a combination of the stereo image. The disparity map is pre-calculated with stereo matching algorithms. We use a depth-aware superpixel segmentation method with an additional *depth term*. A hill-climbing [19] superpixel segmentation method is applied to the color image with a *regularization term* to reduce the complexity. Sparse depth results produced by fast matching algorithms can still be the input, as we utilize random sampling to omit the effect of outliers during plane-fitting. The boundaries of the segmentation are further updated after plane functions have been assigned to each segment. The superpixels are the basic elements of the mapping process. We extract the vertices of each plane and then convert them into the camera coordinate system. For existing 3D planes, we aggregate those whose spatial relationship are planarity while minimizing the total energy function.

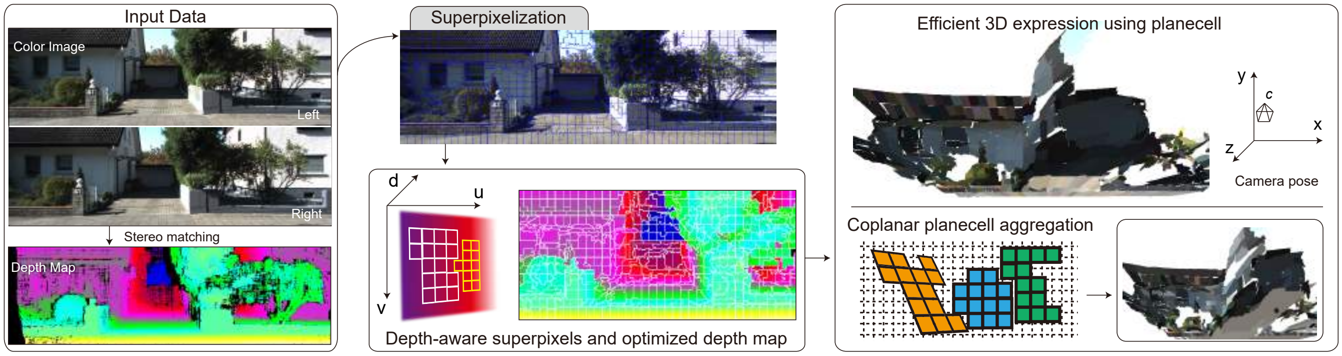

Input Data
Color Image
Left
Right
Stereo matching
Depth Map
Superpixelization
d
u
v
Depth-aware superpixels and optimized depth map
Efficient 3D expression using planecell
y
c
x
z
Camera pose
Coplanar planecell aggregation

Fig. 2. Overview of our 3D representation method: planecell.

## IV. Representing the 3D Space with Planecells

The planecell is the basic unit representing geometric information of objects in the proposed 3D map representation method. Each planecell is a combination of pixels from the color image and uses a joint plane function to deliver their positions. The shape of each planecell is a polygon, which enables us to define their boundaries by vertices. The planecells are adopted by two main processes of stereo matching and superpixel segmentation. For the method is not sensitive to the choice of 3D knowledge acquisition, i.e., various matching algorithms meet its demands or even depth acquisition devices like LiDARs and Kinects, we omit introducing the stereo matching step.

### A. Planecells Extraction

We utilize superpixel segmentation methods to adopt basic planecells from the color reference image. For a color image $\mathcal{I}$, the superpixel segmentation $S = \{S_1, ..., S_k\}$ has the following properties

$$S_i \cap S_j = \emptyset \quad and \quad \mathcal{I} = \bigcup_{Si \in S} \tag{1}$$

Throughout the literature, various superpixel algorithms are graph-based methods that aggregate similar pixels belonging to the same object. This property helps to distinguish planes in the region of interest initially, and superpixel segmentation leaves no holes in the input reference image, which also benefits the 3D map representation. However, the boundary of each superpixel is always unnecessarily heavy and complicated, especially for urban scenarios with structural objects, which increases the computational and storage demands. To address this issue, we propose an improved superpixel method suited our approach based on the prior work [19]. For the hierarchical structure, the segmentation equally divides $\mathcal{I}$ with large to small blocks as each layer. By merging blocks from large to small, i.e., the hill-climbing method, gives the final segmentation and saves block movements. We define the unit of boundary update as a block $\mathrm{b}$. In practice, the size of $\mathrm{b}$ is related to the layer level $l$ of the hill-climbing algorithm and the number of superpixels $|S|$, which is set as $\lfloor\sqrt{\frac{\mathcal{I}}{|S|}}\rfloor/l$ pixels.

At level $l$ of the hill-climbing process, the algorithm proposes a new partitioning $S_l$ with blocks changing to its neighboring superpixel horizontally or vertically. In our implementation, to adopt more efficient segmentation, the boundary block alters its label at level $l$ depending on the costs defined below:

$$\begin{aligned} c(S_i, b_j^l) = &\sum_q \min\{h_{S_i}(q), h_{b_j^l}(q)\} - \alpha_{reg}\|\mu_{S_i} - b_j^l\|_2^2 \\ &- \alpha_{depth}(\theta_{S_i}(b_j^l) - \bar{d_{b_j^l}})^2 \end{aligned} \tag{2}$$

where $h$ is the histogram, $S_i$ is the neighboring segment of block $b_j^l$ and $\bar{d_{b_j^l}} = \frac{\sum_{p \in b_j^l} d_p}{|b_j^l|}$. The block $b_j^l$ will merge to adjacent segment $S_i$ with smallest $c(S_i, b_j^l)$. As the minimum updating unit of our algorithm is at the block level, we iterate changing the boundaries with the smallest blocks until a valid image partitioning is obtained or the maximum run-time is reached.

### B. Superpixel Energy Function

The superpixel segmentation is bounded by the maximization of the energy function, which is defined as the sum of three terms. The energy comprises a color term $H(S)$ based on the histogram of the color space, a regularization term $R(S)$, and a depth term $D(S)$:

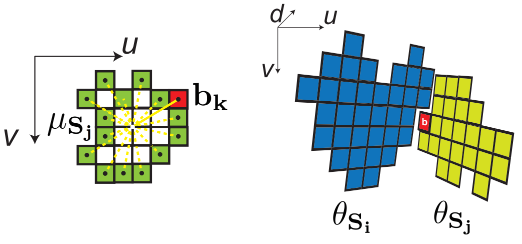
$$E(S) = H(S) + \lambda_{reg}R(S) + \lambda_{depth}D(S) \tag{3}$$

where $\lambda_{reg}$ and $\lambda_{depth}$ are two balancing parameters.

**Color term:** The color term $H(s)$ measures the color distribution of the superpixels and inclines toward superpixels with color histograms that drop into similar bins. With the image segmentation $S$, the color term is formulated as

$$H(S) = \sum_{S_i}\sum_q^Q h_{S_i}(q)^2 \tag{4}$$

where $q$ denotes the histogram bin and $h(.)$ is the number of pixels in the bin. It is not difficult to infer that $H(S)$ reaches its maximum if and only if each histogram is placed in the same bin. Nonetheless, the quality of this evaluation of color is related to the bin size, i.e., the sensibility of color declines when the number of neighboring colors in a single



(a) Regularization term $R$ (b) Depth term $D$

Fig. 3. Beyond color distribution term, we add two terms according to the location and depth of blocks which are the minimum changing units during boundary update.

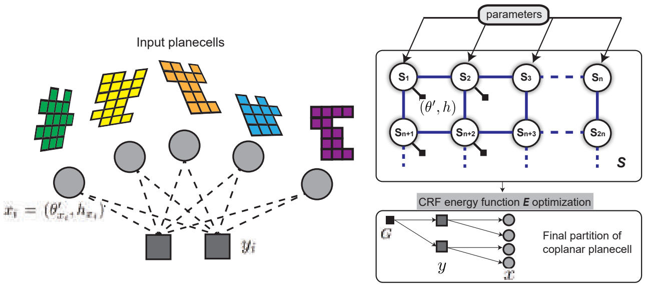

Input planecells
$x_i = (\theta'_{x_i}, h_{x_i})$
$y_i$
parameters
S1
S2
S3
Sn
$(\theta', h)$
Sn+1
Sn+2
Sn+3
S2n
S
CRF energy function E optimization
G
y
x
Final partition of coplanar planecell

Fig. 4. The CRF model.

bin is large.

**Regularization term:** The regularization term $R(S)$ (see Fig. 3(a)) constrains the superpixels to be standard, encouraging straight boundaries. Let $\mu$ be the center of segments or blocks, and $\mathcal{B}_i$ be the set of boundary blocks of segment $i$. The regularization term is given by

$$R(S) = \sum_{S_i} \sum_{b_k \in \mathcal{B}_i} \|\mu_{S_i} - \mu_{b_k}\|_2^2 \tag{5}$$

The value of $R(S)$ is maximized when all blocks on the boundary have the same distances to the neighboring superpixels.

**Depth term:** The depth term (see Fig. 3(b)) comes into effect after the plane function of each segment has been obtained. We denote the plane function of $S_i$ as $\theta_{S_i}$ which equals $(A_i, B_i, C_i)$. The depth distance between a block and neighboring segments is estimated by measuring the difference in the average block depth and the estimated depth generated by the plane function. The term is defined as

$$D(S) = \sum_{S_i} \sum_{b_k \in \mathcal{B}_i} \|\bar{d}_{S_i} - \hat{d}(\mu_{b_k}, \theta_{S_i})\| \tag{6}$$

By applying this term, the segmentation outperforms the former method when the color loses its effect.

### C. Plane Function Estimation

After importing the disparity image, we assign each pixel a label to distinguish whether it is an outlier, i.e., the unmatched pixels. To further identify mismatched pixels, we estimate the plane function by random sampling. The plane-fitting terminates when the number of inliers reaches a target percentage. The difference between the estimated disparity of pixel p with segment $i$ and the input disparity is measured as $\theta_{S_i}(\mathrm{p})$. If this term exceeds a given threshold, the pixel is considered to be an outlier. If no appropriate function can be obtained after a designated number of iterations of $Si$, we omit this segment and re-estimate it after the boundary is updated with depth.

## V. 3D Map Expression

After producing the partition results with a plane function assigned to each superpixel, we extract the vertex of each segment. The vertex set is defined as $v_{S_i}$ to each segment $S_i$. Storing vertices requires much less memory and computation time than all of the plane pixels or edges during 2D-3D conversion.

### A. 2D-3D Conversion

The conversion is based on the vertices. Each vertex set $v_{S_i}$ contains $N$ vertices, and $v_{S_i}$ is composed of variables describing their location in the 2D image and the disparity value estimated with the plane function. Then, for $v_{S_i}$ to segment $S_i$, the position in the 3D coordinate system of the left camera can be calculated using the camera's intrinsic parameters and the relative rotation and translation matrix between the stereo camera. We denote the plane function as $\theta'$ after converting to the real world coordinate. Note that the 2D-3D conversion does not cause loss of precision to each pixel.

### B. Coplanar Planecells Aggregation with CRF

The process of aggregating coplanar planecells starts from a 3D reconstruction result of our proposed method. The target is to assign planecells with a common label if they fit into a similar geometric primitive in the 3D world. This aggregation reveals higher-level comprehension of the environment, which can be further used in the road extraction and understanding of structures. Prior methods utilizing CRF to merge pixels in the 3D world for the purpose of surface segmentation do not integrate existing knowledge of the color image sufficiently, which requires significant computational resources, especially when dealing with large-scale maps.

The plane-based map is demonstrated by a set of discrete planecells $\{x_0, x_1, \ldots, x_n\}$. The process is then presented as a labeling problem from the CRF model $G$, i.e., assign each unit $x_i$ a label $y_i$ whose value indicates the most probable surface to which it belongs. We denote a tuple $(\theta', h)$ to describe a plane, where $\theta'$ is the plane parameters in the camera coordinates and $h$ is the color distribution descriptor. The CRF model is shown in Fig. 4. The implementation of our process merges coplanar units into a larger surface iteratively until each surface is denoted with a unique tuple. The CRF model is defined as $G = (\mathbf{x}, \mathbf{y}, \mathbf{c})$ where $\mathbf{x}$ is the set of nodes denoting the planecells, $\mathbf{y}$ is the labels of $\mathbf{x}$, and $\mathbf{c}$ is the set of boundaries between each adjacent units.

The CRF function is then formulated as the following

$$F = \sum_{x_i} \delta_i(y_i) + \sum_{x_i} \sum_{x_j \in \mathcal{N}_{x_i}} \phi_{i,j}(y_i, y_j) + \sum_{x_i} \sum_{c_i \in \mathcal{B}_{x_i}} \psi_i(c_i) \tag{7}$$

where $\mathcal{N}_{x_i}$ is the neighboring plancells of $x_i$, the potential $\delta_i$ evaluates the color distribution of $x_i$ from the reference image using the histogram of the color space, the pairwise potential $\phi_{i,j}$ measures the difference of depth in the 3D space, which encourages neighboring planecells to belong to the same surface if they are close in both geometric position and pose, and the term $\psi_i$ technically encodes the boundaries of $x_i$. These potentials are further explained in the following.

The term $\delta_i$ is a unary potential that measures the similarity of the planecell unit and the surface with respect to the color histogram:

$$\delta_i(y_i) = \begin{cases} f(S_{x_i}, S_{x_j}) & y_i = y_j \\ 0 & y_i \neq y_j \end{cases} \tag{8}$$

where $f(.,.)$ is the color similarity measurement. This term $\sum_{x_i} \delta_i(y_i)$ increases with planecells share a common label similar in appearance. The potential $\phi_{i,j}(y_i, y_j)$ is designed to constrain the geometric information, which refers of the $\theta'_i$ in each planecell

$$\phi_{i,j}(y_i, y_j) = \theta'_i(S_{x_j})^2 + \theta'_j(S_{x_i})^2 \tag{9}$$

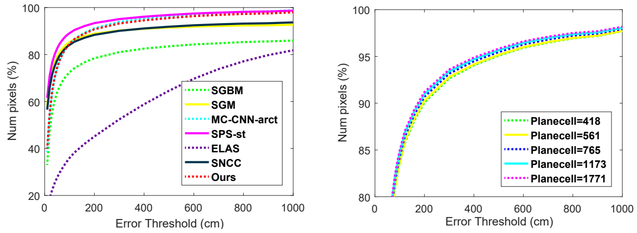
where $\theta'(.)$ denotes the distance value with the plane with function $\theta'$. Let $\theta'_i = (A'_i, B'_i, C'_i)$ be the plane pose of unit $x_i$. The 3D point $p$ in unit $x_i$ obeys $\theta'_i \cdot p = 0$. The potential $\phi_{i,j}$ reaches its maximum when two units agree in their poses. For the potential $\psi_i$, the formula can be written as

$$\psi_i(c_i) = \theta'_i(c_i)^2 \tag{10}$$

The maximization of Eq. 7 is an NP-hard problem solving a CRF model with various variables. We implement the labeling process using a circular greedy algorithm, which merges units within a given range of variation to the greatest extent. Note that the variation range determines the possibility of planarity between adjacent planecells. The algorithm starts from randomly given plancells limited initial labels. By iteratively change the label of each planecell, the algorithm try to aggregate as many planecells into the same surface as possible. Plancells will give a new label once it do not agree with any other plane units.

## VI. EXPERIMENTAL RESULTS

We evaluate our algorithm on two main datasets, namely the KITTI stereo dataset and KITTI odometry dataset [12]. The KITTI stereo dataset separates the images into training and testing sets. The training part includes LiDAR ground truth data with and without occlusions. Each group of images contains two continuous stereo pairs with scene flow information. The outdoor scene dataset provided by the KITTI benchmark is quite challenging, as it contains significant depth variation. Our method shows its advancement dealing with KITTI datasets, whose images are largely of man-made environments that exhibits geometric structures. To better demonstrate the superiority for handling large-scale inputs, we test our algorithm on the KITTI odometry dataset, which has continuous stereo pairs with camera poses.



(a) Pixel-level accuracies on the KITTI stereo dataset with non-occlusion grount truth

(b) The accuracy variance with modification of planecells

Fig. 5. Reconstruction accuracy plots for KITTI stereo datasets in (a). Reconstruction accuracy plots in (b) with different number of planecells.

The results are discussed in terms of accuracy, speed, memory requirements, and the ability to represent useful information. We compare our reconstruction accuracy with the dense point cloud method which directly converts 2D pixels into the 3D world for which exhibits the highest point-level accuracy compared to other representations. We also analyze the 3D map results with a voxel-grid based method. Detailed baselines and evaluations are given in the following.

### A. Implementation Details

As the goal of the proposed method is a new representation of 3D geometric information, we give each planecell an average RGB value for reference. For $\lambda_{reg}$ and $\lambda_{depth}$, we assign them with values according to the initial superpixel size. The plane function is obtained during plane-fitting, and this process may fail if the input depth information is insufficient. In our experiments, the average rate at which the plane function successfully defines all superpixels is $99.95\%$ with the input depth map from SGM. Those planes without plane functions are mostly the area of the sky or reflective objects that will not be converted into the final output. The input depth maps to point or voxel-based 3D space representation methods in our experiments are all calculated by deriving SGM [15]. All experiments in this paper only occupy a single core.

### B. Baselines

We compare our results with several state-of-the-art stereo matching algorithms on the point cloud 3D map reconstructed from them. The method [24] proposed by Zbontar et al. is a preprocessing step for many stereo algorithms, which utilized a convolutional neural network to calculate the matching cost between patches. The corresponding algorithm named MC-CNN-arct outperforms other approaches on both KITTI and Middlebury stereo datasets. We also compare our results with the matching algorithm of Yamaguchi et al. [23] called SPS-st, whose formulation is based on a slanted plane

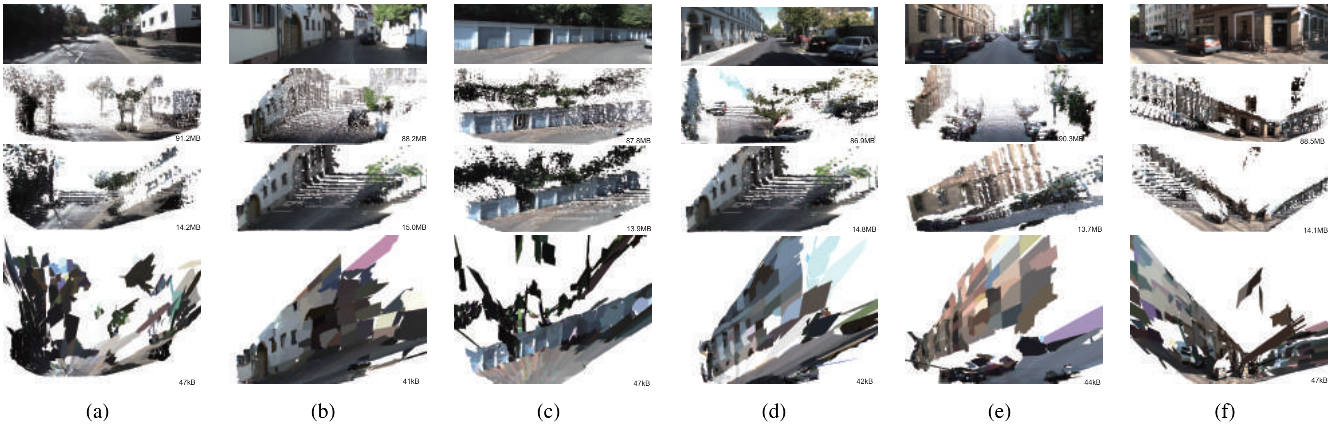


Fig. 6. The 3D space representation with point-based, voxel-based and our planecell methods on KITTI stereo dataset with one frame. Each group of images contains the input, the point-cloud map, the voxel grid map and the planecell map. The size of each representation is also displayed.

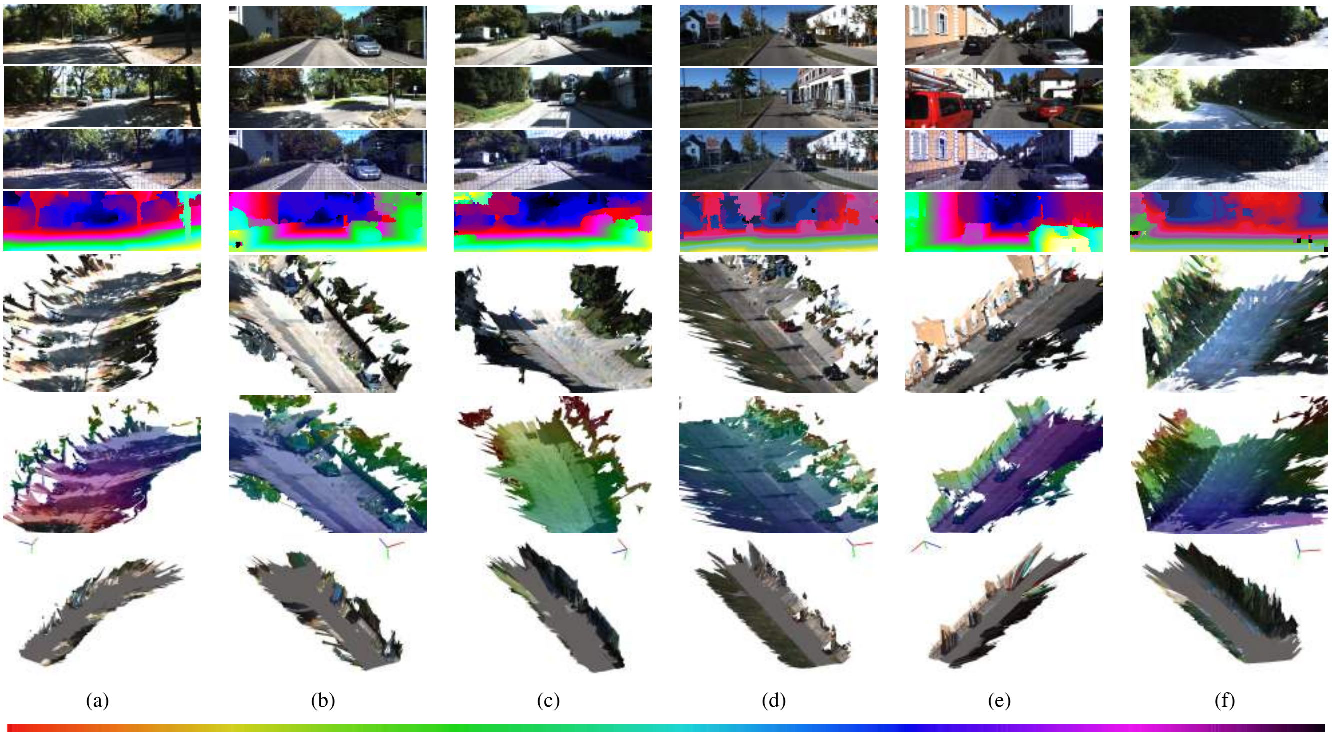


Fig. 7. The intermediate steps and reconstruction results of our planecell method from 50 continuous image sequences on KITTI odometry datasets. Each group of images contains the first input, the last input, the partition, the depth map, the planecell map, the height map and the road extraction. The color bar demonstrating height from low to high is shown at the bottom.

model. Besides, we also test algorithms including SNCC [8], ELAS [13], SGBM and SGM [15] are also listed in our experiments. The SNCC [8] is implemented with additional left-right consistency check and median filters.

The voxelized representation [7], [17] is well developed recently for it standardizes the observations of the regions in space. By following this concept, we implement it by dividing the space into 3D voxel grid. The input contains a depth map and a color reference image. The color estimated for each voxel is the average over the observed pixels. The voxel size in our experiments is fixed to $10cm$ for KITTI datasets in our experiments.

### C. Evaluation and Discussion

We first evaluate reconstruction accuracy by comparing each method to ground truth. The sum of per-pixel Euclidean distance errors over the ground truth is computed after reprojecting into the coordinate of left camera. The comparison demonstrates the pixel-level accuracy of our method. The comparison results are displayed in Fig. 5 (a). We set the parameters of SPS-st to produce $1000$ superpixels. Our method generates nearly $765$ planecells for KITTI dataset referring to the images sizes. And it can be observed from Fig. 5 that more than $80\%$ points of our results are located within $1m$ around the ground truth. The end of each curve is

| Method | $< 10cm$ | $< 20cm$ | $< 50cm$ | $< 100cm$ |
|---|---|---|---|---|
| SGBM | 33.49% | 50.23% | 72.91% | 80.84% |
| SNCC | 57.72% | 67.06% | 79.91% | 85.42% |
| LiDAR Data | 67.56% | 75.16% | 85.21% | 90.73% |

TABLE I
THE RESULTS ON THE KITTI STEREO DATASET BY CHANGING THE INPUT DEPTH MAP.

also restricted to the density of each method. The proposed method achieves comparable pixel-level accuracy compared to dense point clouds reconstructed from state-of-the-art stereo matching algorithms. Different from point clouds containing millions of discrete points holding back further applications, the proposed method is reconstructed based on planes. In our experiments, the output file of point cloud maps is approximately 200 times larger than the proposed method.

For the quality of our results depends on the input depth map during plane-fitting, we then test with different inputs in Table. 1. The ground truth from LiDAR can produce planecell model as well. The loss of precision with ground truth inputs is mainly due to inaccurate superpixelization. The result on the LiDAR input also prove the ability of proposed method to deal with sparse inputs. Another test focusing on changing the number of planecells is shown in Fig. 5 (b). It demonstrates that the precision increases slightly with more planecells, which is due to the probability of better partition of boundaries. The increase of planecells promotes the performance especially in a complicated scene and also raise the computation complexity. The two terms *depth term* and *regularization term* also help improve superpixel segmentation.

We provide several results from three different 3D space representations in Fig. 6. The input depth maps are all generated by SGM [15] method. As shown in Fig. 6(b) and (e), both point-based and voxel-based results become sparse when the disparities grow, mainly because that far scenes do not have sufficient informations. The proposed planecell method avoids this bad influence by summarizing pixels into a 3D plane which restricts blank area in the output. With the *regularization term* $R$, the partition of our method reduces the complexity of boundaries. The boundaries of each planecell influence both following computation time and storage by defining the vertices. For the input depth maps generated by SGM [15] are not full-dense and include many unmatched areas, the proposed method derives the slanted-plane model to produce optimized depth maps. The planecell also benefits the distance measurements during applications like obstacle avoidance. For instance, denote a position in the 3D world as $(x_0, y_0, z_0)$, the shortest distance to planecell with $\theta' = (A', B', C')$ can be calculated as $|\frac{A'x_0+B'y_0-z_0+C'}{\sqrt{A'^2+B'^2+1}}|$. The proposed method also shows advantages for storing the reconstruction results efficiently. In contrast to point-based method saving all locations, the proposed method requires an average of 45kB per frame. The memory requirements of other methods are shown in Fig. 6. The result demonstrates that the proposed method can achieves accurate pixel-level map while reducing the output size.

More detailed results on consecutive frames from KITTI odometry datasets are displayed in Fig. 7. The reconstruction is based on 50 frames with ground truth poses. The map is reconstructed by mapping each frame data to the first left camera coordinate. Moreover, to show the ability of height perception, we color the placecell with an additional height attribute (see the fifth row of Fig. 7). The height is an essential variable for path-planning of autonomous driving. With the proposed CRF model, we further aggregate coplanar planecells and present the road extraction result. As demonstrated in the last row of Fig. 7, the road planecells are given the same color. The road extraction presented on our planecell proves the extensibility for applications in real scenarios.

## VII. CONCLUSION

We propose a novel approach in this paper representing the structural 3D space with basic units of planes named planecell. The planecells are extracted with a depth-aware manner and can be further aggregated if they belong to the same surface applying proposed CRF model. The experiments demonstrate that our method gives consideration to pixel-level accuracy while efficiently express locations of similar pixels. The results avoid the redundancy of point cloud map and limit output map sizes for further applications. In our future work, we plan to import more complex plane models, like spheres and cylinders, to suit more conditions. We also believe that giving each planecell a semantic label would extend the understanding of the environment in a more effective way.

## REFERENCES

[1] M. Agrawal and L. S. Davis. A probabilistic framework for surface reconstruction from multiple images. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 2, pages II–II. IEEE, 2001.

[2] R. Bhotika, D. J. Fleet, and K. N. Kutulakos. A probabilistic theory of occupancy and emptiness. In *European conference on computer vision*, pages 112–130. Springer, 2002.

[3] A. Bódis-Szomorú, H. Riemenschneider, and L. Van Gool. Fast, approximate piecewise-planar modeling based on sparse structure-from-motion and superpixels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 469–476, 2014.

[4] A. Bódis-Szomorú, H. Riemenschneider, and L. Van Gool. Superpixel meshes for fast edge-preserving surface reconstruction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2011–2020, 2015.

[5] A.-L. Chauve, P. Labatut, and J.-P. Pons. Robust piecewise-planar 3d reconstruction and completion from large-scale unstructured point data. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 1261–1268. IEEE, 2010.

[6] A. Dame, V. A. Prisacariu, C. Y. Ren, and I. Reid. Dense reconstruction using 3d object shape priors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1288–1295, 2013.

[7] J. S. De Bonet and P. Viola. Poxels: Probabilistic voxelized volume reconstruction. ICCV, 1999.

[8] N. Einecke and J. Eggert. A two-stage correlation method for stereoscopic depth estimation. In *Digital Image Computing: Techniques and Applications (DICTA), 2010 International Conference on*, pages 227–234. IEEE, 2010.

[9] F. Fraundorfer, K. Schindler, and H. Bischof. Piecewise planar scene reconstruction from sparse correspondences. *Image and vision computing*, 24(4):395–406, 2006.

[10] D. Gallup, J.-M. Frahm, and M. Pollefeys. Piecewise planar and non-planar stereo for urban scene reconstruction. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 1418–1425. IEEE, 2010.

[11] D. Gallup, M. Pollefeys, and J.-M. Frahm. 3d reconstruction using an n-layer heightmap. In *Joint Pattern Recognition Symposium*, pages 1–10. Springer, 2010.

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

[13] A. Geiger, M. Roser, and R. Urtasun. Efficient large-scale stereo matching. In *Asian conference on computer vision*, pages 25–38. Springer, 2010.

[14] C. Hane, C. Zach, A. Cohen, R. Angst, and M. Pollefeys. Joint 3d scene reconstruction and class segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 97–104, 2013.

[15] H. Hirschmüller. Stereo processing by semi-global matching and mutual information. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Citeseer, 2007.

[16] J. Liu, J. Wang, T. Fang, C.-L. Tai, and L. Quan. Higher-order crf structural segmentation of 3d reconstructed surfaces. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2093–2101, 2015.

[17] A. Osman Ulusoy, M. J. Black, and A. Geiger. Patches, planes and probabilities: A non-local prior for volumetric 3d reconstruction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3280–3289, 2016.

[18] S. Sinha, D. Steedly, and R. Szeliski. Piecewise planar stereo for image-based rendering. 2009.

[19] M. Van den Bergh, X. Boix, G. Roig, B. de Capitani, and L. Van Gool. Seeds: Superpixels extracted via energy-driven sampling. In *European conference on computer vision*, pages 13–26. Springer, 2012.

[20] V. Vineet, O. Miksik, M. Lidegaard, M. Nießner, S. Golodetz, V. A. Prisacariu, O. Kähler, D. W. Murray, S. Izadi, P. Pérez, et al. Incremental dense semantic stereo fusion for large-scale semantic scene reconstruction. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 75–82. IEEE, 2015.

[21] C. Vogel, K. Schindler, and S. Roth. 3d scene flow estimation with a piecewise rigid scene model. *International Journal of Computer Vision*, 115(1):1–28, 2015.

[22] T. Whelan, H. Johannsson, M. Kaess, J. J. Leonard, and J. McDonald. Robust real-time visual odometry for dense rgb-d mapping. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 5724–5731. IEEE, 2013.

[23] K. Yamaguchi, D. McAllester, and R. Urtasun. Efficient joint segmentation, occlusion labeling, stereo and flow estimation. In *European Conference on Computer Vision*, pages 756–771. Springer, 2014.

[24] J. Zbontar and Y. LeCun. Stereo matching by training a convolutional neural network to compare image patches. *Journal of Machine Learning Research*, 17(1-32):2, 2016.

[25] M. Zollhöfer, M. Nießner, S. Izadi, C. Rehmann, C. Zach, M. Fisher, C. Wu, A. Fitzgibbon, C. Loop, C. Theobalt, et al. Real-time non-rigid reconstruction using an rgb-d camera. *ACM Transactions on Graphics (TOG)*, 33(4):156, 2014.