GroundNet: Segmentation-Aware Monocular Ground Plane Estimation with Geometric Consistency

Yunze Man, Xinshuo Weng, Kris Kitani
Carnegie Mellon University
5000 Forbes Ave, Pittsburgh, PA, USA
yman, xinshuow, kkitani@andrew.cmu.edu

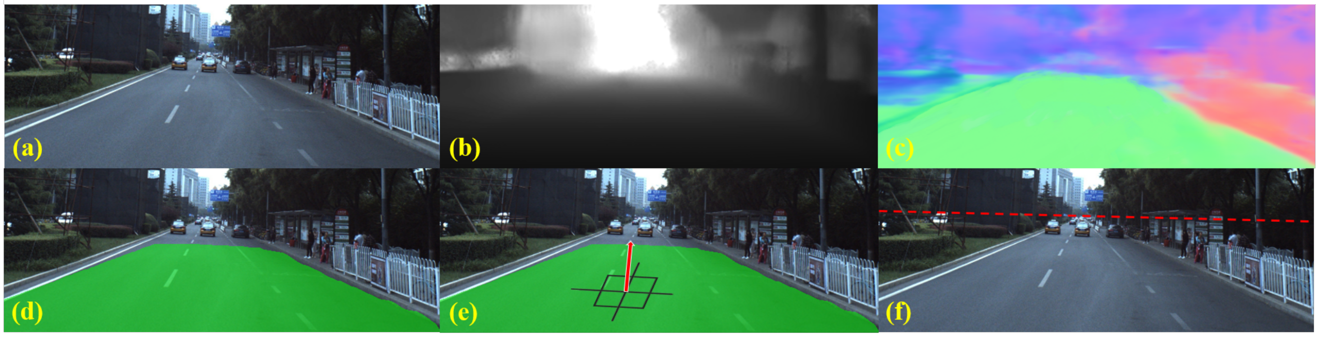

Figure 1. Results from Our approach for ground normal estimation: (a) Input image; (b)(c) geometrically-consistent depth and normal estimation; (d) ground segmentation; (e) final ground normal estimation; (f) computed horizon line from the estimated ground normal.

Abstract

*We focus on the problem of estimating the orientation of the ground plane with respect to a mobile monocular camera platform (*e.g.*, ground robot, wearable camera, assistive robotic platform). To address this problem, we formulate the ground plane estimation problem as an intermingled multi-task prediction problem by jointly optimizing for point-wise surface normal direction, 2D ground segmentation, and depth estimates. Our proposed model – GroundNet – estimates the ground normal in two streams separately and then a consistency loss is applied on top of the two streams to enforce geometric consistency. A semantic segmentation stream is used to isolate the ground regions and are used to selectively back-propagate parameter updates only through the ground regions in the image. Our experiments on KITTI and ApolloScape datasets verify that the GroundNet is able to predict consistent depth and normal within the ground region. It also achieves top performance on ground plane normal estimation and horizon line detection.*

1. Introduction

Understanding the orientation of the ground plane through the use of various visual cues is an important preprocessing step for mobile platforms such as ground robots, assistive robotics and wearable camera systems. An accurate estimate of the ground plane can serve as an important prior for many perception and planning tasks, *e.g.*, tracking [21], semantic segmentation [1], free space estimation [32], 3D object detection [8][23], camera placement estimation [49], 3D reconstruction [17], and scene analysis [16][15]. While many sensors can be used to directly estimate the ground plane (*e.g.*, depth camera, stereo camera, laser scanner), we are primarily interested in innovating ground plane estimation algorithms for mobile platforms that are equipped with only a monocular camera.

Perhaps the most classical approach to ground plane estimation makes use of multi-view geometry or motion cues to first triangulate points in 3D, or directly obtain the 3D point cloud using depth sensors like LIDAR. Then a large plane is fitted to the 3D points using a robust model fitting algorithm like RANSAC [26]. When only a single image is available, parallel lines detected on the ground plane can be used to estimate vanishing points and the horizon line [14]. While these geometry-based approaches are exact with noiseless input, their performance is highly dependent on the reliability of low-level computer vision algorithms to extract corner points or line segments. The current best practice is to use a combination of geometry-based and learning-based methods to obtain both accurate and robust results.

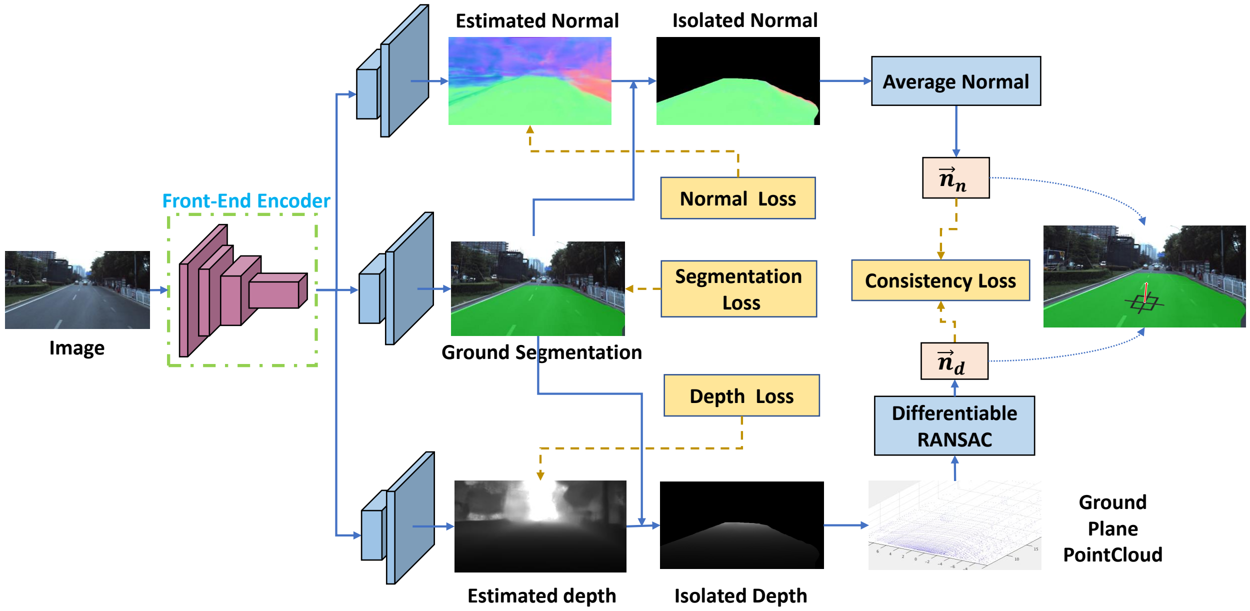

Figure 2. GroundNet architecture. A single image is fed into the front-end encoder and then outputs three heads, namely normal estimation, depth estimation and ground segmentation. The estimated ground segmentation is used to isolate the ground regions and are used to selectively back-propagate parameter updates only through the ground regions in the image. **Depth head:** The ground plane normal $\vec{n}_d$ is obtained by fitting a plane from the ground plane point cloud using the Differentiable RANSAC (DSAC). **Normal head:** We calculate the ground plane normal $\vec{n}_n$ by taking the average over the estimated pixel-wise ground plane normals.

An alternative approach that can be used to estimate the ground plane is the use of monocular surface normal estimation [2]. The basic idea of this approach is to use small image patches to estimate the distribution of surface normals using local visual information. Most prior work in this area are tailored to indoor scenes and are typically not designed to deal with the outdoor scenes involving diverse objects (*e.g.*, vehicles) and natural 'stuff' (*e.g.*, vegetation, sky) with dynamic motion. Also, by design, these methods represent the normals in a pixel-wise manner. They typically do not take into account global priors like the fact that ground plane is often a flat and smooth surface. For these reasons, the estimated normals often suffer from heavy local noise.

To leverage the advantages of both approaches, we propose GroundNet (Figure 2). GroundNet estimates the ground normal in two streams, namely a normal estimation stream and a depth estimation stream. In the first normal estimation stream, we implement pixel-wise normal estimation architecture to obtain a dense normal map. Using the output of the semantic segmentation network, the orientation of the ground plane is computed from the average of the normal estimates identified as ground. In the second depth estimation stream, we obtain the 2.5D depth map by applying a monocular depth estimation network. A point cloud of the ground plane is obtained by back-projecting the pixels belonging to the ground plane based on their depth value. To fit the ground robustly and differentiably in the point cloud, we use Differentiable-RANSAC (DSAC) [4].

In order to remove the effect caused by diverse objects on or nearby the ground in the outdoor scenes, a semantic segmentation stream is used to isolate the ground regions. Semantics labels are used to selectively back-propagate parameter updates only through the ground regions in the image. The backbone network for all three networks is shared: ground segmentation, depth estimation and pixel-wise normal estimation.

As one might expect, the estimated ground plane normals in two streams are not typically not the same. We enforce equivalence by introducing a consistency loss, which minimizes the angular difference between the estimated ground plane orientation from two streams. This supervision signal is flown back to the dense normal estimation network and depth estimation network such that a consistent geometry relationship is explicitly learned in two streams. We argue that this consistency constraint can resolve the ambiguity, to some extent, existing when estimating either the depth or normal alone from a single image.

The contribution of our work is threefold: (1) we propose to segment out the ground area from the estimated normal and depth during back-propagation using semantic segmentation to prevent the adverse effects of irrelevant objects in ground plane estimation; (2) a consistency loss is introduced such that a consistent geometry relationship is explicitly learned between two streams of ground normal estimation; and (3) extensive experiments demonstrate the effectiveness of our method on both ground plane normal estimation and horizon line detection.

2. Related Work

Ground Plane Normal Estimation. Existing approaches for ground plane estimation can be classified into geometry-based methods and learning-based methods.

Geometry-based methods often extract the 3D scene structure (*e.g.*, using multi-view cues, motion cues or depth sensors) and then the ground plane is fitted to the 3D points using a robust model fitting algorithm like RANSAC. [26] identifying the ground using the 3D point cloud from LIDAR. [29] obtains the video frame rate depth maps from the time-of-flight (TOF) cameras and exploits 4D spatiotemporal RANSAC for ground plane estimation. [37] generates the 3D point cloud under a stereo setup and then estimates the ground plane by disparity. Assuming the scene is static, simultaneous localizing and mapping (SLAM) and structure from motion (SFM) approaches can also be used for extracting the 3D scene structure [39][27][36][31][52][47], making ground plane estimation possible.

When only a single image is available, parallel lines detected on the ground plane can be used to estimate vanishing points and the horizon line [14]. Also, [28] shows how grouping detected line segments into quadrilaterals can be used to find orthogonal planes. However, these geometry-based methods are highly dependent on the reliability of low-level computer vision algorithms (*e.g.*, planar homography estimation, line segment detection), especially when given only a single image.

The secondary category methods focus on applying machine learning technique to estimate the ground plane normal either directly or implicitly from other related tasks. There are only a few prior works are direct methods. [12][13] learn a classifier to classify local planar image patches and their orientations first. Then a Markov random field (MRF) model is learned to segment the image into dominant plane segments. [30] achieves the ground plane recognition by learning the lighting-invariant texture feature using regularized logistic regression model. However, these methods make use of shallow learning model and do not benefit from the recent significant progress in deep learning. Also, they do not model the geometric relationship existing in the image. Our proposed method also falls into this category. To the best of our knowledge, this is the first work of direct ground plane estimation which leverages the strong capacity of the deep neural network and the geometry consistency.

Also, one can estimate the ground plane implicitly by solving a related task. One such example of task is 3D surface layout recovery [17][50][53]. These methods are capable of creating a simple indoor layout reconstruction from a single image and then the ground plane normal is able to be estimated. Another example task could be monocular surface normal estimation [35][7][2][3][6][24][43][42], which usually formulate the problem as a dense pixel-wise prediction problem and learn a feed-forward deep neural network classification model. On top of the estimated pixel-wise normals, one can group the pixels with similar normals into the dominant planes and then compute the average normal for each plane. Although these methods achieve significant progress, most of them are tailed only for the indoor scene. In addition, since they do not explicitly estimate the plane normal, the fact that the ground plane is often a flat and smooth surface is ignored. In contrast, we parameterize the output of our methods to be a planar surface normal explicitly and in the meantime leverage the successful architecture design from the surface normal estimation methods.

Self-Supervised Learning via Geometric Consistency Loss. Geometric consistency is proved to be a useful and free supervision signal in many tasks. [51] learns to predict dense flow field between different instances of the same category object consistently across the synthetic and real domain. [34] jointly optimizes the 3D surface point positions and normals to be consistent with the observed light refraction effect. [20] learns a keypoint detector, which is consistent across different viewpoints, by using the epipolar constraints as the free supervision signal. [5] learns a depth estimator under a stereo setting by enforcing estimated depth in two views to be consistent with the disparity. Also, the consistent camera pose estimator and 3D shape predictor are learned via the multi-view projection consistency loss in [40]. [9] learns a 3D geometrically-consistent feature map for reconstruction, segmentation and classification from multi-view observations. [25] enforces the estimated ego-motion consistent with the computed motion using 3D (iterative closest point) ICP on the estimated depth between two frames. Perhaps [45][33][46] are closest to our work in this aspect, which learn to predict the geometrically consistent depth and normal. It has been shown in prior works that the depth and normal are complementary and thus jointly optimizing the two with consistency constraints can improve the performance on both tasks. In our proposed method, we also apply this consistency loss. Different from prior works, the consistency loss is applied to the estimated ground plane normals from two streams (*i.e.*, the normal estimation stream and depth estimation stream).

3. Problem Definition

Given a monocular RGB image as input, our goal is to estimate its ground plane. The plane is represented by its normal vector $\vec{n}$.

Pinhole Camera Model In this problem, the pinhole camera model is adopted. Assume $[u_i, v_i]^T$ is the location of a 2D point P_i on the image plane. Let its corresponding location in the camera 3D coordidate be $Q_i = [x_i, y_i, z_i]^T$, where z_i is the depth. Based on pinhole camera model, we have

$$\begin{aligned} x_i &= (u_i - c_x) \times z_i / f_x, \\ y_i &= (v_i - c_y) \times z_i / f_y, \end{aligned} \tag{1}$$

where c_x and c_y are coordinates of the principle point. f_x and f_y are the focal length along the x and y directions respectively. It can be further written in homogeneous coordinate as

$$P_i = \begin{bmatrix} \lambda u_i \\ \lambda y_i \\ \lambda \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x & 0 \\ 0 & f_y & c_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix} = K_c Q_i \tag{2}$$

where K_c is intrinsic matrix (projection matrix). The intrinsic matrix of the camera is assumed to be known, so that we can get 3D point cloud from the estimated depth map. Therefore, the estimated normal vector $\vec{n}$ is in RGB camera coordinate.

Parameterization The normal vector $\vec{n}$ can also be represented as roll θ and pitch ψ with respect to the up-axis. (θ, ψ) can be further parameterized into horizon line representation (θ, ρ), which allows us to compare our method with horizon line estimation works (see Sec. 5.2).

4. Method

In this section, we describe in detail the proposed GroundNet for ground plane estimation. We first present an overview of the whole GroundNet pipeline and then introduce the two processing streams of the GroundNet. Finally, we illustrate the normal consistency part and the inference scheme for the overall network.

4.1. Approach Overview

Fig 2 depicts the whole framework of the proposed ground plane estimation network (GroundNet). GroundNet consists of three main components. First, a front-end fully convolutional encoder outputs two main estimation heads as well as a ground segmentation mask. Second, each of the two main heads — densely estimated depth map Z and normal map N — goes through an estimation process to get its ground normal. Finally, a consistency loss is added between these two estimated ground normals to further improve both normal and depth estimation as well as the final normal estimation. The input of GroundNet is a monocular RGB image for both training and testing phase, and the final output is the estimated ground normal. In the testing phase, the final ground normal is the average of the two regularized normal vectors from the two streams.

4.2. Normal Estimation Streams

Using VGG16-based [38] front-end encoder, we obtain the dense depth map Z and normal map N for ground plane estimation. Then two separate streams — both guided by the ground plane segmentation mask M — each estimates a ground plane normal.

Depth Stream. For depth estimation, we first use ground segmentation mask to remove the non-ground region from the whole depth map and get the ground region depth map $\hat{Z}$,

$$\hat{Z} = Z \odot M$$

The $\hat{Z}$ contains only ground region depth. Afterwards, following Eq. 1, we back-project each pixel in $\hat{Z}$ to its 3D coordinate (x_i, y_i, z_i), since we have intrinsic matrix K_c and depth z_i for each pixel.

After back-projection, we obtain the point cloud C $(n \times 3)$ for ground regions. In order to get the ground normal from the point cloud, we can use a plane fitting algorithm $f(\cdot)$ to get the ground normal $\vec{n}_d$

$$\vec{n}_d = f(C)$$

This function can be either least square module, as used in [33], or RANSAC-based methods [4]. The RANSAC method tends to perform better because of its insensitivity to outliers. However, it is more time-consuming compared with the simple least square alternative. The depth stream finally outputs a ground normal $\vec{n}_d$ calculated from the estimated depth map.

Normal Stream. Generic surface normal estimation in outdoor scenes suffers from diverse objects with dynamics motion. Thus similar to depth stream, the ground mask is used to segment out the ground region from the whole normal map.

$$\hat{N} = N \odot M$$

The ground region normal map $\hat{N}$ contains normal for all ground pixels. We then make an assumption that local noise in the densely estimated normal fits a Gaussian distribution. This assumption means all the normals in $\hat{N}$ satisfy a Gaussian distribution with its mean μ being the real plane normal. This is similar to giving all normals the same weights during the final normal decision. Then by taking the average of all values in $\hat{N}$, we can get the ground normal $\vec{n}_n$

from normal estimation

$$\vec{n}_n = \frac{1}{M}\sum_{i}^{M}\hat{N}_i$$

where M is the total number of pixels.

4.3. Geometric Consistency

The two normal estimation streams mentioned above output two ground normals, $\vec{n}_d$ and $\vec{n}_n$. As depth and normal are geometrically consistent with each other [33]. These two normals, originated exactly from the depth and normal map respectively, should theoretically be of the same value. However, generated by two separate learning streams without any constraints on the close underlying geometry relationship, they are different in value. Therefore, we enforce the consistency between them by adding a consistency loss $\mathcal{L}_{con}$ on top of the two streams.

$$\mathcal{L}_{con} = \arccos\frac{\vec{n}_d \cdot \vec{n}_n}{\|\vec{n}_d\| \cdot \|\vec{n}_n\|} \tag{3}$$

4.4. Loss Functions

We now explain the loss functions associated with our GroundNet. For pixel i, we denote the estimated depth map for the ground region and ground truth as $\hat{Z}_i$ and Z_i^{gt} respectively. Similarly, we denote the same classes for the normal map as $\hat{N}_i$ and N_i^{gt}.

Our overall loss function is the weighted sum of all four loss terms

$$\mathcal{L}_{GroundNet} = \mathcal{L}_{depth} + \mathcal{L}_{normal} + \eta\mathcal{L}_{seg} + \lambda\mathcal{L}_{con} \tag{4}$$

where η and λ are hyper-parameters to controls the relative importance of the loss terms. The depth loss $\mathcal{L}_{depth}$ is expressed as

$$\mathcal{L}_{depth} = \frac{1}{M}\sum_{i}\|\hat{Z}_i - Z_i^{gt}\|_2^2$$

The surface normal loss $\mathcal{L}_{normal}$ is

$$\mathcal{L}_{normal} = \frac{1}{M}\sum_{i}\|\hat{N}_i - N_i^{gt}\|_2^2$$

And for ground segmentation task, we use the common softmax cross-entropy loss. Our GroundNet is trained by back-projection in and end-to-end manner.

5. Experiments

In this section, we evaluate the performance of ground plane estimation network by conducting extensive evaluations (see Sec. 5.5 and 5.6) on two public benchmark dataset, KITTI and ApolloScape (see Sec. 5.1 and 5.3). A thorough derivation of our evaluation metrics is provided in Sec. 5.2. Furthermore, we perform an in-depth ablation study (see Sec. 5.7) to evaluate the performance of our method. Additional details about our network architecture and training procedure are reported in Sec. 5.4.

5.1. Datasets

KITTI is a famous and popular outdoor autonomous driving dataset [41], in which disparity depth and road semantic labels are provided for a subset of the dataset. We adopt the split scheme proposed by Eigen et al [11], which splits the total 56 scenes from raw KITTI dataset into 28 for training and 28 for testing. The ground truth of the ground plane normal is calculated from the given extrinsic matrix.

ApolloScape is a big autonomous driving dataset for scene parsing [19], instance segmentation and self localization. It contains a great number of image frames with complete depth information and scene labels. It contains 40,963 images for training and 8330 images for validation. Similar to KITTI, we calculate the ground truth normal for the ground plane from the provided calibration matrix.

5.2. Evaluation Metrics

Because many previous works proposed to directly estimate horizon line from outdoor images, we compare our method not only with normal estimation methods, but also horizon line estimation methods. We give proof that under a certain assumption, ground normal is equivalent to horizon line of an image.

Normal estimation To compare with normal estimation methods, we evaluate the angular error between ground truth normal $\vec{n}_{gt}$ and estimated ground normal $\vec{n}$ in terms of degree, as used in [10].

Horizon line estimation To compare with horizon line estimation methods, we change our parameterization from normal vector $\vec{n}$ into (θ, ρ) and report the error of these two parameters, as mentioned in [44, 18], where θ is the angle the horizon line makes with the horizontal axis, and ρ is the perpendicular distance from the principal point to the horizon line.

As shown in Figure 3, the ground plane can either be represented by its normal $\vec{n}$ or be represented by its roll θ and pitch ψ to the horizontal plane, as long as the camera center and principle point is known — Namely, given the intrinsic matrix, we can transform our estimated normal into horizon line without losing any information.

From Eq. 2 we know, a point in camera coordinate, Q_i, is related to a point in image coordinate P_i as follows:

$$P_i = [\lambda u_i, \lambda v_i, \lambda]^T = K_c Q_i \tag{5}$$

where K_c is intrinsic matrix. Therefore, $K_c^{-1}P_i = Q_i$. Let P and Q be sets of points on the horizon line in camera and

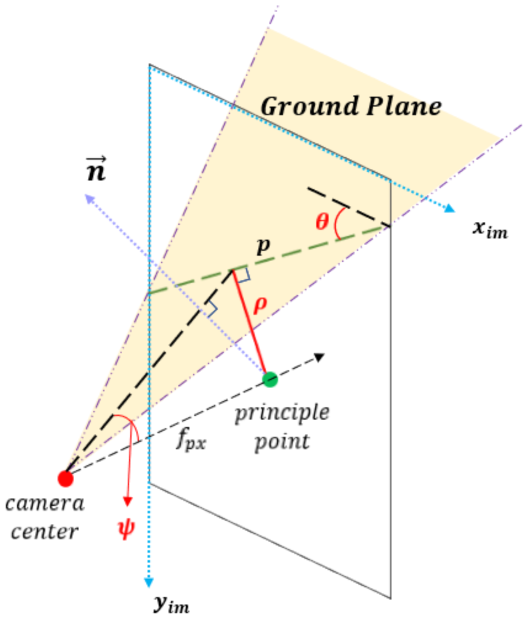

Figure 3. Illustration of our geometric model. Ground normal $\vec{n}$ can be transformed into horizon line in the image, which is parameterized as roll θ and pitch ρ, where $\rho = f_{px} \cdot tan\psi$.

image coordinate, respectively. Then,

$$Q^T = P^T K_c^{-T} \tag{6}$$

The normal of a plane is perpendicular to all in-plane vectors. Because horizon line is in ground plane, therefore, in camera coordinate, $\vec{n} \perp Q$, or $Q^T \vec{n} = 0$. Therefore, from equation 6:

$$P^T K_c^{-T} \vec{n} = 0 \tag{7}$$

Let $K_c^{-T} \vec{n} = A = [a, b, c]^T$, the horizon line in image coordinate can be represented as:

$$P^T A = 0 \tag{8}$$

$$ax + by + cz = 0 \tag{9}$$

Let $z = 1, a/c = a_c, b/c = b_c$,

$$a_c x + b_c y + 1 = 0 \tag{10}$$

Assuming that the positive x-direction is to the right, the positive y-direction is down and principle point is (W_p, H_p), we can calculate the horizon line (θ, ρ) parameters as:

$$\theta = \arctan -\frac{a_c}{b_c}$$

$$\rho = -\frac{|a_c W_p + b_c H_p + 1|}{\sqrt{a_c^2 + b_c^2}}$$

By these derivations, the estimated ground normal $\vec{n} = [n_x, n_y, n_z]^T$ can be transformed as (θ, ρ) horizon line representation in image coordinate.

5.3. Data Augmentation

For both KITTI and ApolloScape, the cameras are fixed to the car, which means the ground plane orientation does not change significantly over time. As a result, the ground plane normals lack variety in the dataset. Thus, we manually introduce random rolls and pitches to the dataset by performing rotation and vertical translation to the images.

Specifically, for roll, we randomly rotate the images around their principal points within a certain limit. For pitch, we randomly translate the images up or down within a certain limit. Afterward, the images are cropped according to the principle point to get rid of black margins. This method allows us to increase the plane normal variety without introducing significant distortion or inaccuracy to the datasets.

Moreover, we set three different rotation limits, namely 5°, 10° and 15°, in order to determine the influence of this augmentation. Similarly, we set the limit of vertical translation to be 0.05, 0.1 and 0.15 image units.

5.4. Implementation Details

We implement GroundNet using the publicly available Tensorflow framework. The front-end convolutional encoder of GroundNet uses VGG-16 [38] as backbone. We initialize the front-end encoder with network pre-trained on ImageNet. Because the ground segmentation task is simple, for now, we pre-train it with direct ground label supervision separately and fix its weights afterward. Adam optimizer [22] is used to optimize the network on KITTI dataset. The initial learning rate is 1e-4, (β_1, β_2) is set to $(0.9, 0.999)$. For ApolloScape dataset, we initialize with model pre-trained on KITTI, and fine-tune with learning rate set to 5e-5.

5.5. Ground Normal Evaluation

The quantitative and qualitative results for the ground plane normal estimation are shown in Table 1 and Figure 4. We compare our methods with two state-of-the-art surface normal estimation methods designed for the indoor scenes: Marr SkipNet [3] and GeoNet [33]; and a direct ground plane normal estimation method: HMM [10].

It is clear that GroundNet significantly outperforms all three baselines under all kinds of dataset augmentation (including no augmentation). We argue that this is because our proposed method for direct ground plane normal estimation leverages strong capacity of the deep neural network compared to HMM (baseline direct method). Also, our method is able to deal with the diverse objects in the outdoor scenes by introducing the ground segmentation network.

Table 1. Ground normal evaluation results on KITTI and ApolloScape. (5°, 0.05) means the dataset is augmented by random adding roll and pitch within 5 degree and 0.05 image units, respectively. *: Note that the result of [10] is obtained from the original published paper without any dataset augmentation, it only reports evaluation results on KITTI.

	error / deg						
	KITTI				**ApolloScape**		
	0°, 0	5°, 0.05	10°, 0.1	15°, 0.15	5°, 0.05	10°, 0.1	15°, 0.15
Marr SkipNet [3]	-	8.32	9.73	10.41	7.98	9.86	10.95
GeoNet [33]	-	6.96	8.08	9.57	6.56	7.71	8.56
HMM [10]*	4.10	-			-		
GroundNet (Ours)	**0.96**	**3.01**	**4.47**	**6.52**	**2.97**	**4.39**	**5.66**

Table 2. Ablation results on KITTI and ApolloScape with dataset augmentation. We show the results of the estimated ground plane normal from the normal stream and depth stream with the use of the estimated ground segmentation. Results from the depth stream with the use of ground truth segmentation mask is also shown.

	error / deg					
	KITTI			**ApolloScape**		
	5°, 0.05	10°, 0.1	15°, 0.15	5°, 0.05	10°, 0.1	15°, 0.15
normal stream + estimated seg	6.73	7.99	9.35	5.78	7.47	8.24
depth stream + estimated seg	3.01	4.47	6.52	**2.97**	4.39	5.66
depth stream + groundtruth seg	**2.99**	**4.45**	**6.51**	**2.97**	**4.34**	**5.62**

Table 3. Horizon line evaluation results on KITTI. (5°, 0.05) means augmentation limits as mentioned in Table 1. The unit for θ and ρ are degree and 10^{-2} image unit.

	5°, 0.05		10°, 0.1		15°, 0.15	
	θ	ρ	θ	ρ	θ	ρ
Zhai et al. [48]	3.36	3.9	4.37	4.9	5.99	6.2
DeepHorizon [44]	2.26	3.7	4.12	4.7	5.93	5.6
GroundNet	**2.25**	**2.5**	**3.31**	**3.2**	**5.11**	**4.1**

Table 4. Horizon line evaluation results on ApolloScape.

	5°, 0.05		10°, 0.1		15°, 0.15	
	θ	ρ	θ	ρ	θ	ρ
Zhai et al. [48]	2.58	2.9	3.94	3.7	5.79	5.1
DeepHorizon [44]	**1.98**	3.2	3.42	3.3	4.43	3.9
GroundNet	2.14	**2.7**	**3.28**	**2.9**	**4.16**	**3.8**

5.6. Horizon Line Evaluation

Following the derivation in Sec. 5.2, we are able to convert our estimated ground plane normal to the horizon line given the intrinsic matrix. We thus compare our method with two state-of-the-art horizon line estimation methods: Zhai et al. [48] and DeepHorizon [44].

Quantitative results on KITTI and ApolloScape are shown in Table 3 and 4 respectively. We also plot the qualitative results shown in Figure 5. We show that our proposed GroundNet outperforms all other baseline methods on two datasets under all kinds of dataset augmentation. Interestingly, the performance of DeepHorizon is close or sometimes slightly better to GroundNet when the dataset augmentation is small (*i.e.*, 5°, 0.05). However, when we increasing the amplitude of the augmentation, it is clearly to see that the performance of the GroundNet outperforms the DeepHorizon. This shows that our proposed method is more robust to the diversity of the rolls and pitches in the input data. Also, we notice that the average performance of all methods on ApolloScape dataset is better than on KITTI. This is because the images from the ApolloScape dataset are collected on the main road and thus contains less complex scenarios than KITTI.

5.7. Ablation Study

In order to justify the effect of different model components, we conduct ablation experiments for GroundNet. The results are shown in Table 2.

First, we can see that the performance from the normal stream with the help of the estimated segmentation slightly is better results than GeoNet (2nd row in Table 1). This shows that our joint training with the consistency loss helps improve the performance for each single stream. Also, by comparing the 1st and 2nd rows in Table 2, it is shown that estimating the ground plane normal from the depth stream is easier than the normal stream when applying the same estimated segmentation results. Moreover, the results (in the 2nd and 3rd rows of Table 2) show that the performance of the depth stream is almost the same when provided with either estimated segmentation or groundtruth segmentation. This demonstrates that the estimated segmentation from our segmentation network is able to provide similar help as the

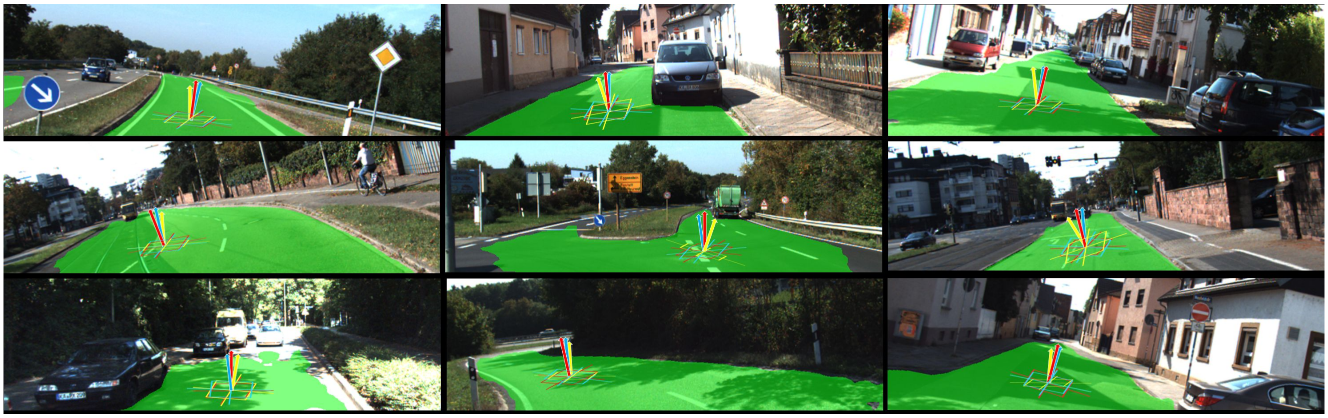

Figure 4. Qualitative results for the ground plane normal estimation. Red: groundtruth. Blue: GroundNet. Yellow: Marr SkipNet. We show that our proposed GroundNet consistently outperforms all other baselines.

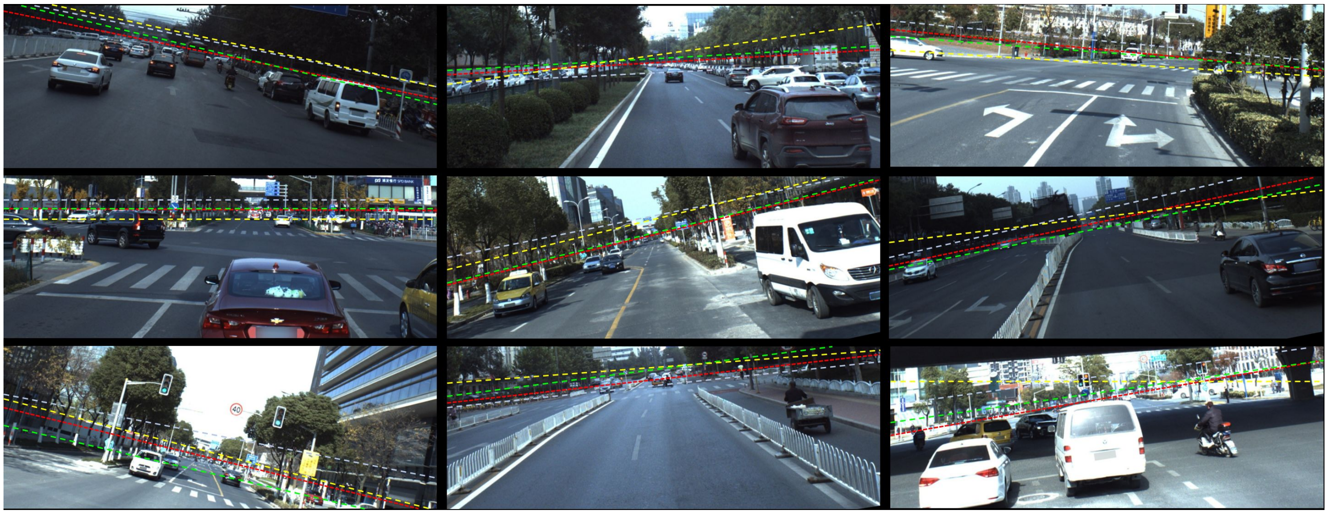

Figure 5. Qualitative results for the horizon line estimation. Red: groundtruth. Green: GroundNet. Yellow: DeepHorizon. Gray: Zhai et al. We show that our proposed GroundNet consistently outperforms all other baselines.

groundtruth segmentation.

6. Conclusion

In this paper, we propose GroundNet for ground plane estimation from a single image. GroundNet leverages the advantages from both geometry-based and learning-based methods. In other words, GroundNet benefits from the successful deep neural network architecture design for monocular surface normal estimation and depth estimation. Also, a geometric consistency loss is applied to the estimated ground plane normal between the normal stream and depth stream such that GroundNet is able to predict consistent depth and normal. In addition, we show the effectiveness of the segmentation stream in terms of removing the effect caused by irrelevant objects in the outdoor scenes. Experimental results on KITTI and ApolloScape datasets show that the proposed method performs favorably against state-of-the-arts for both ground plane normal estimation and horizon line estimation.

References

[1] J. M. Alvarez, T. Gevers, Y. LeCun, and A. M. Lopez. Road scene segmentation from a single image. *ECCV*, 2012. 1

[2] A. Bansal, X. Chen, and B. Russell. PixelNet: Representation of the pixels, by the pixels, and for the pixels. *arXiv*, 2017. 2, 3

[3] A. Bansal, B. Russell, and A. Gupta. Marr Revisited: 2D-3D Alignment via Surface Normal Prediction. *CVPR*, 2016. 3, 6, 7

[4] E. Brachmann, A. Krull, S. Nowozin, J. Shotton, F. Michel, S. Gumhold, and C. Rother. DSAC - Differentiable RANSAC for camera localization. *CVPR*, 2017. 2, 4

[5] G. J. Brostow. Unsupervised Monocular Depth Estimation with Left-Right Consistency. *CVPR*, 2017. 3

[6] S. B. Busam. Predicting depth , surface normals and semantic labels with a common multi-scale convolutional architecture (arXiv 2014) Seminar Recent Trends in 3D Computer Vision. *ICCV*, 2015. 3

[7] W. Chen, D. Xiang, and J. Deng. Surface Normals in the Wild. *ICCV*, 2017. 3

[8] X. Chen and Y. Zhu. 3D Object Proposals for Accurate Object Class Detection. *NIPS*, 2015. 1

[9] R. Cheng, Z. Wang, and K. Fragkiadaki. Geometry-Aware Recurrent Neural Networks for Active Visual Recognition. *NIPS*, 2018. 3

[10] R. Dragon and L. Van Gool. Ground plane estimation using a hidden markov model. *CVPR*, 2014. 5, 6, 7

[11] D. Eigen, C. Puhrsch, and R. Fergus. Depth map prediction from a single image using a multi-scale deep network. *NIPS*, 2014. 5

[12] O. Haines and A. Calway. Detecting planes and estimating their orientation from a single image. *BMVC*, 2012. 3

[13] O. Haines and A. Calway. Recognising Planes in a Single Image. *TPAMI*, 2015. 3

[14] R. Hartley and A. Zisserman. Multiple View Geometry in Computer Vision. *Cambridge University Press*, 2007. 1, 3

[15] V. Hedau, D. Hoiem, and D. Forsyth. Thinking inside the box: Using appearance models and context based on room geometry. *ECCV*, 2010. 1

[16] D. Hoiem, A. Efros, and M. Hebert. Putting Objects in Perspective Derek. *CVPR*, 2006. 1

[17] D. Hoiem, A. A. Efros, and M. Hebert. Recovering surface layout from an image. *IJCV*, 2007. 1, 3

[18] Y. Hold-Geoffroy, K. Sunkavalli, J. Eisenmann, M. Fisher, E. Gambaretto, S. Hadap, and J.-F. Lalonde. A perceptual measure for deep single image camera calibration. *CVPR*, 2018. 5

[19] X. Huang, X. Cheng, Q. Geng, B. Cao, D. Zhou, P. Wang, Y. Lin, and R. Yang. The apolloscape dataset for autonomous driving. *arXiv*, 2018. 5

[20] Y. Jafarian, Y. Yao, and H. S. Park. MONET: Multiview Semi-supervised Keypoint via Epipolar Divergence. *NIPS*, 2018. 3

[21] S. M. Khan and M. Shah. A Multiview Approach to Tracking People in Crowded Scenes Using a Planar Homography Constraint. *ECCV*, 2006. 1

[22] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv*, 2014. 6

[23] J. Ku, M. Mozifian, J. Lee, A. Harakeh, and S. Waslander. Joint 3D Proposal Generation and Object Detection from View Aggregation. *IROS*, 2018. 1

[24] B. Li, C. Shen, Y. Dai, A. Van Den Hengel, and M. He. Depth and surface normal estimation from monocular images using regression on deep features and hierarchical CRFs. *CVPR*, 2015. 3

[25] R. Mahjourian, M. Wicke, and A. Angelova. Unsupervised Learning of Depth and Ego-Motion from Monocular Video Using 3D Geometric Constraints. *CVPR*, 2018. 3

[26] M. W. McDaniel, T. Nishihata, C. A. Brooks, and K. Iagnemma. Ground plane identification using LIDAR in forested environments. *ICRA*, 2010. 1, 3

[27] B. Mičušík and J. Košecká. Piecewise planar city 3D modeling from street view panoramic sequences. *CVPRW*, 2009. 3

[28] B. Micusik, H. Wildenauer, and M. Vincze. Towards detection of orthogonal planes in monocular images of indoor environments. *ICRA*, 2008. 3

[29] F. Mufti, R. Mahony, and J. Heinzmann. Robust estimation of planar surfaces using spatio-temporal RANSAC for applications in autonomous vehicle navigation. *Robotics and Autonomous Systems*, 2012. 3

[30] J. A. J. Osuna-Coutino, C. Cruz-Martinez, J. Martinez-Carranza, M. Arias-Estrada, and W. Mayol-Cuevas. I Want to Change My Floor: Dominant Plane Recognition from a Single Image to Augment the Scene. *ISMAR*, 2016. 3

[31] H. Ovrén and P.-E. Forssén. Spline Error Weighting for Robust Visual-Inertial Fusion. *CVPR*, 2018. 3

[32] D. Pfeiffer and U. Franke. Efficient representation of traffic scenes by means of dynamic stixels. *IEEE Intelligent Vehicles Symposium*, 2010. 1

[33] X. Qi, R. Liao, Z. Liu, R. Urtasun, and J. Jia. Geonet: Geometric neural network for joint depth and surface normal estimation. *CVPR*, 2018. 3, 4, 5, 6, 7

[34] Y. Qian, M. Gong, and Y.-H. Yang. 3D Reconstruction of Transparent Objects with Position-Normal Consistency. *CVPR*, 2016. 3

[35] Z. Ren and Y. J. Lee. Cross-Domain Self-supervised Multi-task Feature Learning using Synthetic Imagery. *CVPR*, 2018. 3

[36] J. L. Schönberger and J.-M. Frahm. Structure-from-Motion Revisited. *CVPR*, 2016. 3

[37] S. Se and M. Brady. Ground plane estimation, error analysis and applications. *Robotics and Autonomous Systems*, 2002. 3

[38] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *ICLR*, 2015. 4, 6

[39] J. P. Tardif, Y. Pavlidis, and K. Daniilidis. Monocular visual odometry in urban environments using an omnidirectional camera. *IROS*, 2008. 3

[40] S. Tulsiani, A. A. Efros, and J. Malik. Multi-view Consistency as Supervisory Signal for Learning Shape and Pose Prediction. *CVPR*, 2018. 3

[41] J. Uhrig, N. Schneider, L. Schneider, U. Franke, T. Brox, and A. Geiger. Sparsity invariant cnns. *3DV*, 2017. 5

[42] P. Wang, X. Shen, B. Russell, S. Cohen, B. Price, and A. L. Yuille. SURGE : Surface Regularized Geometry Estimation from a Single Image. *NIPS*, 2016. 3

[43] X. Wang, D. F. Fouhey, and A. Gupta. Designing Deep Networks for Surface Normal Estimation. *CVPR*, 2015. 3

[44] S. Workman, M. Zhai, and N. Jacobs. Horizon lines in the wild. *BMVC*, 2016. 5, 7

[45] Z. Yang, P. Wang, Y. Wang, W. Xu, and R. Nevatia. LEGO: Learning Edge with Geometry all at Once by Watching Videos. *CVPR*, 2018. 3

[46] Z. Yang, P. Wang, W. Xu, L. Zhao, and R. Nevatia. Unsupervised Learning of Geometry with Edge-aware Depth-Normal Consistency. *AAAI*, 2018. 3

[47] C. Yuan and G. Medioni. 3D Reconstruction of background and objects moving on ground plane viewed from a moving camera. *CVPR*, 2006. 3

[48] M. Zhai, S. Workman, and N. Jacobs. Detecting vanishing points using global image context in a non-manhattan world. *CVPR*, 2016. 7

[49] W. Zhang and J. Kosecka. Extraction , matching and pose recovery based on dominant rectangular structures. *ICCV*, 2003. 1

[50] Y. Zhang, S. Song, P. Tan, and J. Xiao. PanoContext: A whole-room 3D context model for panoramic scene understanding. *ECCV*, 2014. 3

[51] T. Zhou, P. Krähenbühl, M. Aubry, Q. Huang, and A. A. Efros. Learning Dense Correspondence via 3D-guided Cycle Consistency. *CVPR*, 2016. 3

[52] Z. Zhou. Robust Plane-Based Structure From Motion. *CVPR*, 2012. 3

[53] C. Zou, A. Colburn, Q. Shan, and D. Hoiem. LayoutNet: Reconstructing the 3D Room Layout from a Single RGB Image. *CVPR*, 2018. 3