Improved Plane Detection in 3D Reconstruction from 2D Images

Joseph Quan

Abstract

3D reconstruction is a fundamental technology with applications in autonomous driving, virtual reality, and game development. Plane detection is a critical component in 3D reconstruction, as planes form the structural backbone of most environments. However, existing plane detection methods have limitations in their accuracy, while machine learning based plane detection is largely limited by its training dataset. This research aims to enhance plane detection in 3D reconstruction from 2D images by integrating monocular depth estimation, point cloud generation, RANSAC, and clustering techniques to better detect and reconstruct planes. In this project, I generated depth maps from 2D images by implementing monocular depth generation. Next, I then utilized pyRANSAC, a plane detection algorithm to detect basic planes. I then improved the algorithm to detect multiple planes and used a clustering algorithm to address the many artifacts induced by the algorithm. My proposed method was compared with existing approaches to show that my method has a more robust and more accurate plane detection.

Introduction

3D reconstruction creates digital 3D models of real-world objects and scenes from 2D images or other data, which is widely used in different areas, such as automotive driving, virtual reality, robotics for navigation, and game development. In order to reconstruct images in 3D, many methods have been researched, which involved depth sensors, stereo vision, and Structure from Motion (SfM) [16]. Those traditional methods face significant limitations, such as high cost and errors in long-range and outdoor environments. Given the convenience of capturing 2D images and videos, leveraging them for 3D reconstruction is a promising alternative. My research utilizes monocular depth estimation, only needing a single image for 3D reconstruction [1].

Plane detection is a crucial part of 3D reconstruction because it simplifies a scene by identifying flat surfaces, which are prevalent in many environments, like buildings, which enables a more accurate reconstruction with less data. This is especially useful in fields like robotics, autonomous navigation, and virtual reality where understanding the layout of a space is important. However, existing plane detection methods have limitations in their accuracy. For example, machine learning based plane detection is largely limited by its training dataset [15].

One of the most used techniques for robust plane detection is the RANdom SAmple Consensus (RANSAC), which is a global iterative method for estimating the parameters of a certain model from input data points contaminated by a set of outliers (noisy data) [8].

This research aims to enhance plane detection in 3D reconstruction from 2D images by integrating monocular depth estimation, point cloud generation, RANSAC, and clustering techniques to better detect and reconstruct planes. My method utilizes the RANSAC algorithm to detect multiple planes in a point cloud representation of a 2D image, which I improve on using clustering to polish My detected planes.

In this project, I first generated depth maps from 2D images by implementing monocular depth generation. From the depth map, I generated the point cloud 3D reconstruction. I then utilized pyRANSAC, a plane detection algorithm to detect basic planes [2]. I improved the algorithm to detect multiple planes as the original pyRANSAC could only detect one plane. However, there is bleeding between two different planes and streaking of the wrong plane on top of another. A clustering algorithm was used to address this problem [3], refining plane detection by also taking distance between points into account. The proposed method was then compared against existing approaches and evaluated side by side to assess overall accuracy. My results showed more robust and more accurate plane detection.

Related Work

In order to reconstruct images in 3D, many methods have been researched, which involved depth sensors, stereo vision, and Structure from Motion (SfM) [16]. My research utilizes monocular depth estimation, only needing a single image for 3D reconstruction.

Plane detection is a crucial part of 3D reconstruction because it simplifies a scene by identifying flat surfaces, which are prevalent in many environments, like walls, ceilings, tables, and floors, which enable a more accurate reconstruction with less data. This is especially useful in fields like robotics, autonomous navigation, and virtual reality, where understanding the layout of a space is important. There are three main steps to 3D plane detection and reconstruction, with each involving a variety of algorithms: monocular depth estimation to create the depth map (Depth Anything V2 [1], [11]), creating the point cloud, then processing and segmenting the point cloud into planes (RANSAC [8], Hough Transform [9], [12]). Other approaches include using supervised machine learning to create a model that detects planes given an input image, which requires large amounts of labeled data, from datasets such as NYU Depth V2 [5].

To begin, monocular depth estimation aims to infer depth from a single image, allowing generation of a 3D representation (i.e. point cloud) of the scene. Traditional methods use algorithms like Structure from Motion (SfM) and Multi-View Stereo (MVS) [16], which require multiple images to estimate depth by analyzing different views. However, single image approaches need to rely on different methods to analyze and generate depth maps.

One well-known and state-of-the-art monocular depth estimation model is Depth Anything V2 [1], which uses a transformer based architecture trained on extensive depth datasets to predict and create a depth map from a single image. The model operates in three key steps [11]:

1. Train a reliable teacher model based on DINOv2-G purely on high-quality synthetic images.
2. Produce precise pseudo depth on large-scale unlabeled real images.
3. Train final student models on pseudo-labeled real images for robust generalization.

The high accuracy of Depth Anything V2 makes it a valuable tool for monocular 3D reconstruction, allowing us to effectively create disparity maps which can be converted into depth maps for plane detection.

One of the most used techniques for robust plane detection is the RANSAC algorithm [8]. It iteratively fits planes to 3D point clouds by distinguishing inliers and outliers. In each iteration, the algorithm first randomly selects 3 points, fits a plane to those three points, calculates which other points are inliers of the plane by evaluating if the distance between that point and the plane is below a certain threshold. This process is repeated multiple times until the plane with the most inliers is found. This algorithm is widely used due to its robustness against noise but only detects one dominant plane at a time, which will require modifications to detect multiple planes.

Another approach for plane detection is the Hough Transform [9], [12], which converts an image from Cartesian to polar coordinates, representing each point in the image space as a sinusoidal curve in the Hough space. In addition, two points in a line segment generate two curves, which are overlaid at a location that corresponds with a line through the image space. There are fMy main steps in this process: edge detection, transformation into parameter space, voting mechanism, and identifying peaks. First, the image is run through an edge detection algorithm (ex. Canny Edge Detector) to detect features in the image. These image features are then mapped and transformed into the parameter space for further analysis, where each detected feature or point casts a vote for possible plane parameters in an accumulator array. Finally, the plane with the most votes is selected as the best fit plane [9], [12]. The Hough transform is an effective method for detecting multiple planes, however it is computationally expensive for large scale problems.

In addition to RANSAC and Hough Transform, DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a clustering algorithm used to refine plane detection by grouping points while filtering out noise [3], [13]. When applied with RANSAC, DBSCAN can help reduce artifacts, overlapping planes, and outliers, improving accuracy in monocular 3D reconstruction. While effective, DBSCAN is sensitive to parameter changes, and may struggle with varying point densities. However, it is still an important tool that can significantly enhance plane detection robustness.

Other methods for plane detection are based on machine learning, such as PlaneRCNN and PlaneRecNet. PlaneRCNN is a machine learning based method that uses a convolutional neural network to detect and segment planes in images [14]. It used a supervised learning approach that required labeled datasets for training, and enabled the detection of multiple planes from a single image. However, PlaneRCNN had limitations in generalizing to unseen environments and suffered from inaccuracies. A more recent advancement, PlaneRecNet is an improved CNN based model that enhances plane detection by integrating a multi task learning framework [4]. PlaneRecNet [4] integrates a single-stage instance segmentation network for piecewise planar segmentation and a depth decoder to reconstruct the scene from a single RGB image. To achieve this, it uses several novel loss functions (geometric constraint) that jointly improve the accuracy of piecewise planar segmentation and depth estimation. Meanwhile, a novel Plane Prior Attention module is used to guide depth estimation with the awareness of plane instances. Unlike PlaneRCNN, PlaneRecNet improves generalization and reduces artifacts in complex scenes.

In conclusion, monocular 3D reconstruction is a powerful alternative to traditional depth sensing methods, with plane detection being an integral part to accurate reconstruction. Depth Anything V2 improves monocular depth estimation [1], [11]. RANSAC [8], Hough Transform [9], [12], and DBSCAN [3], [13] detect and refine plane segmentation, each with strengths and trade-offs. Deep learning models like PlaneRCNN [14] and PlaneRecNet [4] further enhance detection but require large datasets for training. By integrating monocular depth estimation, geometric methods, and machine learning, I am able to achieve greater accuracy and efficiency in plane detection for 3D reconstruction.

Methodology

1. Monocular Depth Generation

In order to do 3D reconstruction, I need to obtain the depth map of the scene captured by the camera, as shown in Figure 1(a). Instead of using depth sensors, I used monocular depth map generation to get the depth map directly from the 2D images, which is more convenient and easier to use in different conditions. I selected the most advanced monocular depth generation method, the Depth Anything V2 [1] model to generate the disparity map from the 2d images.

2. Point Cloud Generation

In order to generate the 3D point cloud of the scene, I started from the disparity map that I generated above, as shown in Figure 1(a). I first converted the disparity map into the depth map through inputting the focal length of the camera that was used in the NYU depth v2 dataset [5] to take pictures [7]. The focal length information was provided by the NYU depth v2 dataset. Then from the depth map I generate the 3D point cloud using the pinhole camera model [18].

3. Plane Detection

The last stage is to do plane detection from the generated 3D point cloud described above, as shown in Figure 1(b). I started with pyRANSAC-3D [5] to do plane detection. The method could find the main plane in the scene, but it can only find one plane in the reconstructed 3D point cloud. In order to find more planes, I modify the pyRANSAC algorithm to find a set number of planes, from the biggest to smallest, which is limited by the number of points. The modified method worked very well in finding multiple planes. But there were some artifacts found, caused by the detection algorithm bleeding between the detected planes. Later, I used clustering with DBSCAN (Density-Based Spatial Clustering of Applications with Noise) [3] from SKLearn to remove artifacts. The final method worked very well for indoor images, and achieved better results compared to other methods on outdoor images.

4. Deep Learning based Plane Detection

As described above, my plane detection method was based on RANSAC plus the clustering method as the refinement. I also tried some deep learning based plane detection methods. PlaneRecNet [4] is one state-of-art deep learning based method proven to achieve good results in 2D plane detection. I used this model to test the plane detection quality on my test samples, and compared with my method in the following session.

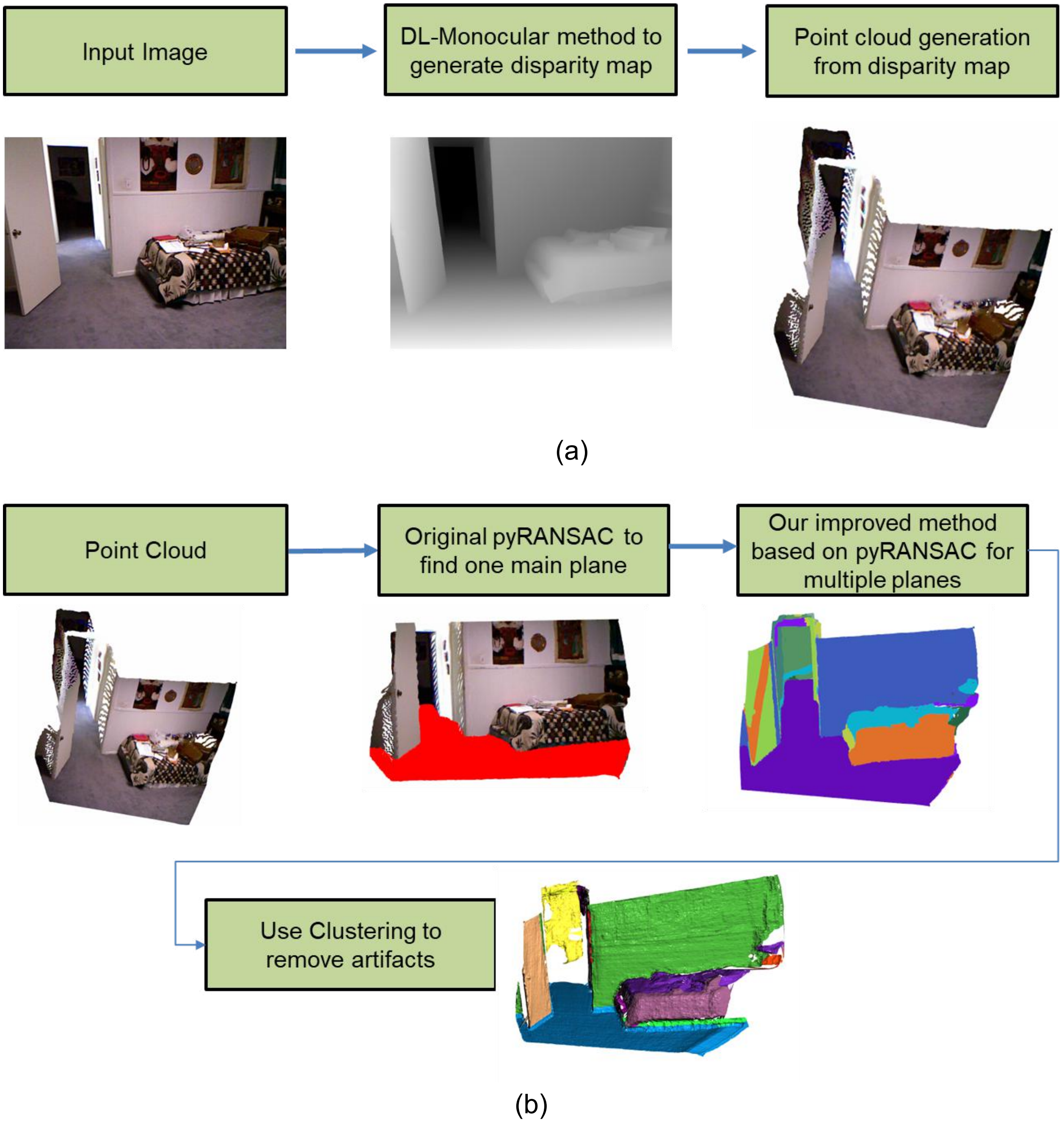

Input Image
DL-Monocular method to generate disparity map
Point cloud generation from disparity map
(a)
Point Cloud
Original pyRANSAC to find one main plane
Our improved method based on pyRANSAC for multiple planes
Use Clustering to remove artifacts
(b)

Figure 1: Graphical description of my methodology. (a) Using an input image to generate the scene disparity map, I then generated the 3D point cloud from the disparity map with focus length information. (b) Plane detection from the 3D point cloud generated from (a). My algorithm detects multiple planes in the scene. The added clustering method refined the results and removed the artifacts.

Experiments and Discussion

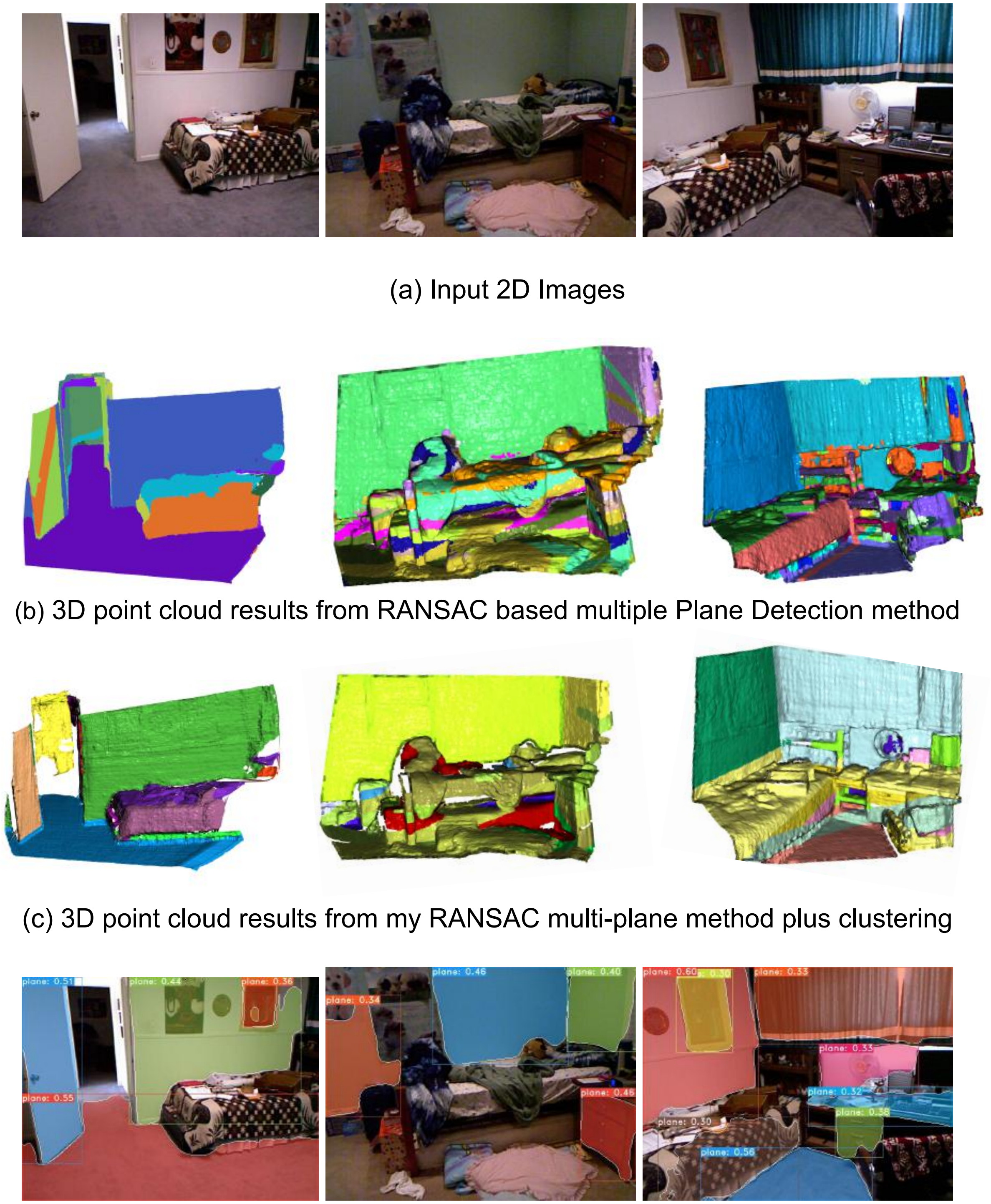
First, the proposed method was tested on the images from the NYU depth v2 Database [5], which is a collection of indoor scenes. 3 examples of the processed results are shown in Figure 2. Figure 2(a) shows the input 2D images. The scenes in the images contain lots of objects with lots of noise, which makes plane detection quite challenging.

Figure 2(b) shows the results of the 3D point cloud obtained from my multiple plane detection method based on RANSAC. Compared to the input 2D images, the generated 3D point clouds very well represented the 3D scene generation from the 2D image scene. In the 3D point cloud, each color plane represents one plane that's detected inside the 3D point cloud. From the examples, you can see that the method showed very high accuracy of the planes in the scene. Although the scenes are complicated, most of the planes in the scenes were able to be detected.

Figure 2(c) shows the results of plane detection after the clustering algorithm [3]. It refined the plane detection results obtained in Figure 2(b), made the plane detection more precise especially on small objects and boundaries. The detected planes very well represented the original structure shown in the input 2D images.

Figure 2(d) shows the results from the deep learning plane detection method PlaneRecNet [4]. Because the method doesn't generate 3D reconstruction. The detected 2D planes were displayed overlapping the original 2D images. The colorful patches representing the planes detected from the method. The results show that only a limited number of planes were detected from PlaneRecNet, much less than those from my results. Also, the precision of the planes are much less than my results. It's hard to reconstruct the original scenes through the detected 2D planes by the deep learning method. One reason for the poor performance is that the scenes are quite complicated with random objects inside the rooms. Another reason is that deep learning based methods largely rely on the training dataset. But generally there is not a robust enough plane detection dataset, especially datasets labeled with ground truth for complicated scenes, that could be used to do plane detection directly from 2D input images. The problem is even more obvious for outdoor images and scenes, since it's much more challenging to get the depth map for outdoor scenes.

From the examples shown in Figure 2, it clearly indicated that my plane detection method based on RANSAC with clustering refinement outperforms both traditional RANSAC methods and even the state-of-art machine learning methods like PlaneRecNet.



(a) Input 2D Images

(b) 3D point cloud results from RANSAC based multiple Plane Detection method

(c) 3D point cloud results from my RANSAC multi-plane method plus clustering

(d) Results from Deep Learning based Plane Detection method PlaneRecNet [4]

Figure 2: Plane detection results from my method and the comparison with the state-of-art deep learning method. (a) Input 2D Images. (b) 3D point cloud results from RANSAC based multiple Plane Detection. (c) 3D point cloud results from my RANSAC multi-plane method plus clustering. (d) Results from Deep Learning based Plane Detection method PlaneRecNet [4].

Conclusion and Future Work

In this project, instead of using depth sensors, I used 2D images directly with monocular depth estimation to generate a 3D model with depth information, which was later used for plane detection.

From the point cloud, I used a RANSAC based method to detect planes in the scene, which was further improved with a clustering strategy to refine results and remove artifacts.

I compared my method with PlaneRecNet, which is a machine learning based method. My method outperforms this machine learning based method because it's not limited by the training dataset, and it does not rely on segmentation, so that it's more robust to work on images of complicated contents, while that is general for normal taken images.

My future work will focus on:

1. Improving the accuracy of point clouds through improving depth map generation, and adjusting the point clouds for more robustness to noise.
2. Further improving the quality of plane detection for complicated scenes and outdoor scenes, which could improve the flexibility of the method to adapt to more environments.
3. Further simplifying the algorithm to have better latency, allowing the method to be used in commercial products.
4. Extend plane detection past planes into other shapes, so that more complicated 3D scenes could be represented with small amounts of data and with higher accuracy.
5. Buildings are made up of many different shapes that are not just expressed in simple planes. Research can be extended to extending my method to complex shapes like spheres or cones.

Bibliography

[1] L. Yang, H. Zhao, B. Kang, Z. Huang, Z. Zhao, X. Xu, and J. Feng, “Depth Anything V2,” arXiv preprint arXiv:2406.09414, 2024. doi:10.48550/arXiv.2406.09414

[2] L. Mariga, “PyRANSAC-3D: A Python tool for fitting primitives 3D shapes in point clouds using RANSAC algorithm,” GitHub, 2022. [Online]. Available: https://github.com/leomariga/pyRANSAC-3D

[3] “DBSCAN (clustering) from scikit-learn,” Scikit-learn documentation. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.cluster.DBSCAN.html

[4] Y. Xi, F. Shu, J. Rambach, A. Pagani, and D. Stricker, “PlaneRecNet: Multi-task learning with cross-task consistency for piece-wise plane detection and reconstruction from a single RGB image,” arXiv preprint arXiv:2110.11219, 2021. doi:10.48550/arXiv.2110.11219

[5] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, “Indoor segmentation and support inference from RGBD images,” in Proc. ECCV, 2012. [Online]. Available: https://cs.nyu.edu/~fergus/datasets/nyu_depth_v2.html

[6] “MiDaS: Monocular depth estimation,” GitHub, 2024. [Online]. Available: https://github.com/isl-org/MiDaS

[7] Q.-Y. Zhou, J. Park, and V. Koltun, “Open3D: A modern library for 3D data processing,” arXiv preprint arXiv:1801.09847, 2018. [Online]. Available: https://arxiv.org/pdf/1801.09847

[8] M. A. Fischler and R. C. Bolles, “Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography,” Commun. ACM, vol. 24, no. 6, pp. 381–395, 1981. doi:10.1145/358669.358692

[9] D. Borrmann, J. Elseberg, K. Lingemann, and A. Nüchter, “The 3D Hough transform for plane detection in point clouds: A review and a new accumulator design,” 3D Research, vol. 2, no. 2, Jun. 2011. doi:10.1007/3dres.02(2011)3

[10] Ai4ce, “PEAC: Fast plane extraction using agglomerative hierarchical clustering (ICRA 2014),” GitHub, 2018. [Online]. Available: https://github.com/ai4ce/peac

[11] NeurIPS, “Depth Anything V2 Poster,” 2024. [Online]. Available: https://neurips.cc/virtual/2024/poster/94431

[12] “Hough Transforms,” ScienceDirect. [Online]. Available: https://www.sciencedirect.com/topics/computer-science/hough-transforms

[13] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, “A density-based algorithm for discovering clusters in large spatial databases with noise,” in Proc. 2nd Int. Conf. Knowledge Discovery and Data Mining (KDD-96), 1996. [Online]. Available: https://www.dbs.ifi.lmu.de/Publikationen/Papers/KDD-96.final.frame.pdf

[14] S. Liu, Y. Chen, T. Wu, S. Han, and Y. Furukawa, “PlaneRCNN: 3D plane detection and reconstruction from a single image,” arXiv preprint arXiv:1812.04072, 2018. [Online]. Available: https://arxiv.org/pdf/1812.04072

[15] N. Silberman et al., “NYU Depth v2 dataset,” 2012. [Online]. Available: https://cs.nyu.edu/~fergus/datasets/nyu_depth_v2.html

[16] M. Kholil, et al., “Structure from Motion and Multi-View Stereo-based 3D reconstruction,” IOP Conf. Series: Materials Science and Engineering, vol. 1073, 2021. doi:10.1088/1757-899X/1073/1/012066

[17] Geiger, Andreas, et al. “The KITTI Vision Benchmark Suite.” IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012.

[18] Richard Hartley and Andrew Zisserman, “Multiple View Geometry in Computer Vision”, Cambridge University Press, 2003.