




# From Depth Sensing to Deep Depth Estimation for 3D Reconstruction: Open Challenges

Charles Hamesse<sup>1,2</sup><sup>a</sup>, Hiep Luong<sup>2</sup><sup>b</sup> and Rob Haelterman<sup>1</sup><sup>c</sup>

<sup>1</sup>*XR Lab, Department of Mathematics, Royal Military Academy, Belgium*

<sup>2</sup>*imec - IPI - URC, Ghent University, Belgium*


**Keywords:** Depth Sensing, Depth Estimation, 3D Reconstruction.


**Abstract:** For a few years, techniques based on deep learning for dense depth estimation from monocular RGB frames have increasingly emerged as potential alternatives to 3D sensors such as depth cameras to perform 3D reconstruction. Recent works mention more and more interesting capabilities: estimation of high resolution depth maps, handling of occlusions, or fast execution on various hardware platforms, to name a few. However, it remains unclear whether these methods could actually replace depth cameras, and if so, in which scenario it is really beneficial to do so. In this paper, we show that the errors made by deep learning methods for dense depth estimation have a specific nature, very different from that of depth maps acquired from depth cameras (be it with stereo vision, time-of-flight or other technologies). We take a voluntarily high vantage point and analyze the state-of-the-art dense depth estimation techniques and depth sensors in a hand-picked test scene, in the aim of better understanding the current strengths and weaknesses of different methods and providing guidelines for the design of robust systems which rely on dense depth perception for 3D reconstruction.


## 1 INTRODUCTION

In recent years, dense depth sensing and estimation techniques have been the subject of significant research efforts. In fact, depth perception is the cornerstone of portable 3D reconstruction systems, which are necessary for numerous robotics applications such as mapping, obstacle avoidance or autonomous navigation. In many cases, being able to perform 3D reconstruction with sensors as small and light as possible is of great interest. To give an example, in various emergency and military contexts, being able to perform 3D mapping to form a clear, up-to-date 3D representation of a given environment is of critical importance: improving the team's situational awareness will help to better execute operations and do better-informed decisions. Also in these cases, it is likely that 3D models will not be readily available, or simply outdated since the event creating the emergency had a direct impact on the 3D environment. Using the traditional rotating LiDAR devices for 3D reconstruction is not possible, as they are still heavy, expensive and can be hard to navigate. Depth or

RGB-D cameras are cheaper and more easily moved around, at the expense of a loss of sensing accuracy and operational range. Technological developments in depth sensing technologies bring depth perception at a small form-factor and acquisition cost thanks to the various technologies behind depth cameras: stereo vision, structured light, time-of-flight or MEMS LiDAR<sup>1</sup> camera. With such cameras, 3D reconstruction is achieved with satisfying accuracy in a range of scenarios, such as reconstructing a static object by rotating smoothly around it, or mapping small scale interior spaces (Zollhöfer et al., 2018). The 3D reconstruction of dynamic large scale scenes, on the other hand, remains the subject of much research (Wang et al., 2021), (Yuan et al., 2022a). Going further, using RGB cameras with a given deep learning depth estimation method would be even more practical, as these cameras can be extremely small and consume little power. The current deep learning literature contains a wide range of algorithms to convert RGB frames to depth maps. Learning-based algorithms keep improving on the task of dense depth estimation based on RGB frames (single-view depth estimation)

<sup>a</sup> <https://orcid.org/0000-0002-2321-0620>

<sup>b</sup> <https://orcid.org/0000-0002-6246-5538>

<sup>c</sup> <https://orcid.org/0000-0002-1610-2218>

<sup>1</sup>Microelectromechanical systems (MEMS) scanning mirrors allow to build quasi-mechanical LiDAR devices with low power and reduced size.

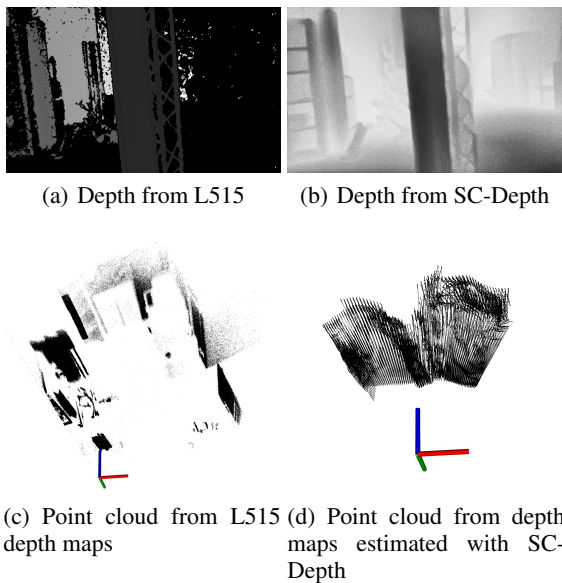


Figure 1: Testing VoxelMap with 10 frames of a sequence in a lab room featuring several closets, various equipment and a central pillar. On the left, we use the depth maps from the Intel Realsense L515. On the right, we use the depth maps estimated with SC-Depth based on the RGB images of the L515.

or sequences (multi-view depth estimation). Yet, we fail to see these algorithms deployed in real-life operational scenarios. To illustrate our point, we show an example execution of the probabilistic mapping system proposed in (Yuan et al., 2022a) running on 10 depth maps acquired with an Intel Realsense L515 RGB-D camera, and 10 depth maps estimated with the state-of-the-art SC-Depth algorithm (Sun et al., 2022) on the RGB frames of that same camera in Figure 1. Clearly, the results are extremely different: the L515 point cloud is relatively sparse but still geometrically correct, but in the case of the deep depth maps, registration simply fails. In fact, the errors present in both of these depth maps are extremely different. Therefore, instead of performing a quantitative analysis of a given technique or sensor as is commonly done in the field, we propose a high-level, qualitative analysis of i) the current state-of-the-art off-the-shelf depth sensing cameras and ii) the latest methods for depth estimation based on images taken with RGB cameras. Our goal is not to compare depth sensors between themselves (see (Zhang et al., 2021), (Zollhöfer et al., 2018) or (Tychola et al., 2022) for comparisons of depth sensors) or depth estimation algorithms (see (Ming et al., 2021), (Dong et al., 2021)), but rather to compare the outputs of both categories in a more practical manner.

The aim of this paper is not to provide a quanti-

tative benchmark or an exhaustive survey, but rather to draw the main characteristics of both categories of methods and their results. Doing so, we hope to give the community useful insight on how to port existing 3D reconstruction systems from RGB-D sensors to RGB cameras with deep depth estimation setups. Our contributions are the following:

1. We propose an overview of the recent portable dense depth sensors;
2. We propose an overview of the recent deep learning-based methods for dense depth estimation;
3. We execute a qualitative evaluation, analyze the common failure cases of the methods in both categories, and discuss potential research directions and implementation designs to alleviate these issues.

## 2 RELATED WORK

We start with an overview of available depth cameras, then proceed to review recent algorithms for depth estimation based on RGB frames or sequences.

### 2.1 Depth Sensing

Depth sensing technologies can be categorized in two main groups: active and passive. Active depth sensing methods include structured light, direct and indirect time-of-flight (ToF). Passive methods include multi-view such as stereo vision, depth from motion, depth from defocus, etc. Recent depth cameras mainly use active and passive stereo as well as time-of-flight, as shown on our list of state-of-the-art sensors in Table 2. Their functioning can be summarized as follows:

- Passive stereo: using two forward-facing cameras, the concept is similar to human binocular vision. Corresponding feature points are found in the image pairs, then the depth of these points can be computed using the known baseline (distance between both cameras) and the coordinate displacement these feature points in both frames. The Zed Mini camera uses this technology (ZED, 2017).
- Active stereo: in addition to passive stereo, a structured light pattern is projected on the scene to help finding corresponding feature points. For example, the Intel Realsense D455 can work with passive or active stereo (Intel, 2020a).
- Indirect time-of-flight: an infrared wave is directed to the target object, and the sensor array detects the reflected infrared component. The depth

Product	Technology	Range [m]	Size [mm]	Weight [g]	Power [W]
Intel Realsense D455	Active stereo	.6 - 6	124 x 26 x 36	390	3.5
Intel Realsense L515	MEMS LiDAR	.25 - 9	61 ( $\varnothing$ ) x 26	100	3.5
Microsoft Azure Kinect DK	Time-of-Flight	.25 - 3	103 x 39 x 126	440	5.9
Zed Mini	Passive stereo	.15 - 24	124 x 30 x 26	63	1.9

Figure 2: Main specifications of commonly used depth cameras. All the cameras listed in this table are tested in this work. Range indicates the operational range given by the manufacturer.

of each pixel is computed using the phase difference between the radiated and reflected wave. One such camera is the Microsoft Azure Kinect V2 (Microsoft, 2020).

- **Direct time-of-flight:** a light emitter is directed towards each point in the field of view of the sensor to emit a pulse, then the depth is computed using time taken for the pulse to come back to the sensor. If using a laser, then these methods are referred to as LiDAR. Directing the emitter to scan the whole field of view of the device can be done in different ways, e.g. with a mechanical rotating device (traditional scanning LiDAR), solid-state or MEMS. The recent Intel Realsense L515 implements the MEMS LiDAR technology (Intel, 2020b).

While all cameras have a similar power consumption, we see clear discrepancies in size and weight, with the Intel Realsense L515 and the ZED Mini being by far lighter than the others. In this work, we will evaluate all of the cameras referenced above.

## 2.2 Depth Estimation

Recent techniques to estimate dense depth maps from RGB images rely on deep learning methods, and more specifically, convolutional layers and Transformer architectures (Vaswani et al., 2017). As always in deep learning research, methods are trained and evaluated on certain datasets. The training dataset may differ from the evaluation dataset. Naturally, the performance of these methods in real-life scenarios will be extremely dependent on the training dataset. Therefore, we start with a brief review of the common datasets for depth estimation, then review the state-of-the-art methods in different depth estimation paradigms.

### 2.2.1 Datasets

The most commonly used datasets for depth estimation are KITTI (Geiger et al., 2012), featuring road scenes, and NYUDepth (Nathan Silberman and Fergus, 2012), featuring indoor scenes similar to those in

which we are interested. The NYU dataset records 464 video sequences with an RGB-D camera (Microsoft Kinect). These video sequences cover a variety of indoor scenes, including living rooms, kitchen, bathrooms. Other important datasets include SUN-RGBD (Song et al., 2015), which aggregates RGB-D images from several other depth datasets (NYU depth v2, Berkeley B3DO (Janoch, 2012), and SUN3D (Xiao et al., 2013)), captured with various depth cameras. In total, SUN-RGBD contains 10 335 images. When developing depth estimation algorithms, researchers use the depth sensed from the depth camera as ground truth.

### 2.2.2 Algorithms

We distinguish pure single-view depth estimation algorithms from algorithms making use of multi-view constraints.

**Single-View Depth Estimation.** State-of-the-art methods in this category include DepthFormer (Guizilini et al., 2022), which builds upon the Transformer (Vaswani et al., 2017) to model the global context with an effective attention mechanism. BinsFormer (Li et al., 2022) also uses a Transformer architecture but formulates depth prediction as classification-regression problem (first predicting probabilistic representations of discrete bins then computing continuous predictions via a linear combination with bins centers). Another state-of-the-art method is NeW-CRF (Yuan et al., 2022b), which leverages Conditional Random Fields (CRFs) in a custom windowed fully-connected manner to speed up computation. All of these methods are trained and evaluated on the NYUv2 and KITTI datasets. In our experiment, we use DepthFormer and BinsFormer.

**Multi-View Depth Estimation.** A major issue with single-view depth estimation is scale ambiguity. Given a 2D RGB image, there is no way the neural network can compute the precise absolute depth. Recent works attempt to correct the scale by using multi-view depth consistency constraints during train-

ing. Current state-of-the-art multi-view depth estimation methods typically require the computation of a multi-view cost-volume, which offers good accuracy but can lead to an important memory consumption and a slow inference. MaGNet (Bae et al., 2022), evaluated on 7-Scenes and ScanNet, aims to reduce the computational cost by predicting a single-view depth probability depth distribution, sampling this distribution then weighting the samples using a multi-view depth consistency constraint. TCMonoDepth (Li et al., 2021), evaluated on NYUv2, enforces multi-view depth alignment constraint during training, but keeps the inference on a single frame. ViDAR (virtual LiDAR) (Guizilini et al., 2022), proposes a new cost volume generation method based on a specific depth-discretized epipolar sampling method. Finally, SC-Depth (Sun et al., 2022) uses image pairs as input and synthesizes the depth for the second view using the predicted depth in the first view and a rigid transformation. In our experiment, we use TCMonoDepth and SC-Depth.

#### Multi-View Depth Estimation with Camera Poses.

Learning-based methods that extend multi-view information with relative camera pose information provided by another system such as a SLAM algorithm or another sensor have been proposed. Since external information is needed to execute these methods, they fall out of the scope of this paper.

### 3 IMAGE FORMATION AND DEPTH MAPS

Camera images are formed by projecting 3D world points to the 2D image plane, then transforming them to the 2D pixel space. The most commonly used camera projection model in computer vision literature is the pinhole model illustrated in Figure 3.

A depth map  $\mathbf{D} \in \mathbb{R}^{H \times W}$ , where  $H$  and  $W$  are the height and width of the image in pixels, contains the depth information of each pixel, i.e. the position of the corresponding 3D point on the forward axis  $z_c$ , starting from the optical center. For a 3D point  $\mathbf{q} = [q_x, q_y, q_z]^T \in \mathbb{R}^3$  and pixel coordinates  $p_x \in [0, W - 1]$ ,  $p_y \in [0, H - 1]$ , we have:

$$D_{p_x, p_y} = q_z \text{ for } \mathbf{q} \text{ s.t. } \mathbf{p} = \pi_{\mathbf{K}}(\mathbf{q}) \quad (1)$$

where  $\pi_{\mathbf{K}}(\cdot)$  is the camera projection operator associated with the intrinsic matrix  $\mathbf{K}$ . This operator and its inverse  $\pi_{\mathbf{K}}^{-1}$  allow to convert depth maps to point clouds and vice versa. For more information on camera models and projections, we refer the reader to (Hartley and Zisserman, 2003).

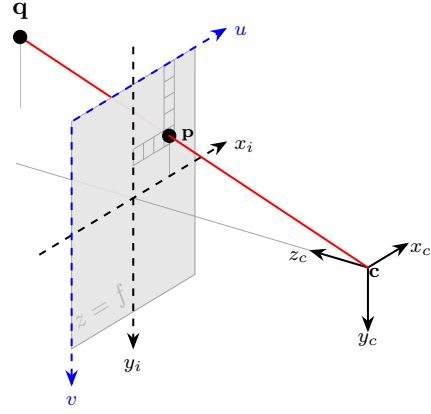


Figure 3: Camera projection model.  $(x_c, y_c, z_c)$  is the camera coordinate system centered around the optical center  $\mathbf{c}$ ,  $(x_i, y_i)$  is the image plane coordinate system, and  $(u, v)$  is the pixel coordinate system. The image plane is at a focal length distance  $f$  of the optical center  $\mathbf{c}$  and orthogonal to the optical axis  $z_c$ .

### 4 QUALITATIVE EVALUATION

We first define the principal criteria to which we will pay attention during our evaluation:

- Point density, which must be high enough for a satisfactory dense 3D reconstruction (this may depend on the target application). It will depend on the sensor resolution, field of view, and depth map density (DMD). We can express the latter as:

$$\text{DMD} = \frac{\# \text{ points}}{H \times W} \quad (2)$$

- Bias and variance, which describe the errors in the depth maps in terms of spread and distance from the correct values;
- Connectivity and presence of ghost structures, which relates how connected surfaces appear connected in the depth maps (and associated point clouds) and the opposite, whether wrong connections between objects or wrong structures are found.

To perform our test, we manually pick an indoor scene, with the only constraints that it should be diverse (with various geometric shapes and textures) and not degenerate (e.g. a flat white wall). We show the scene chosen for our experiment in Figure 5. Since the goal is to complement the typical quantitative evaluations carried out in all research or specifications papers, our evaluation consists in a qualitative, visual inspection of the point cloud resulting from the depth maps. Comparing the point clouds from the same scene, resulting from different sensors

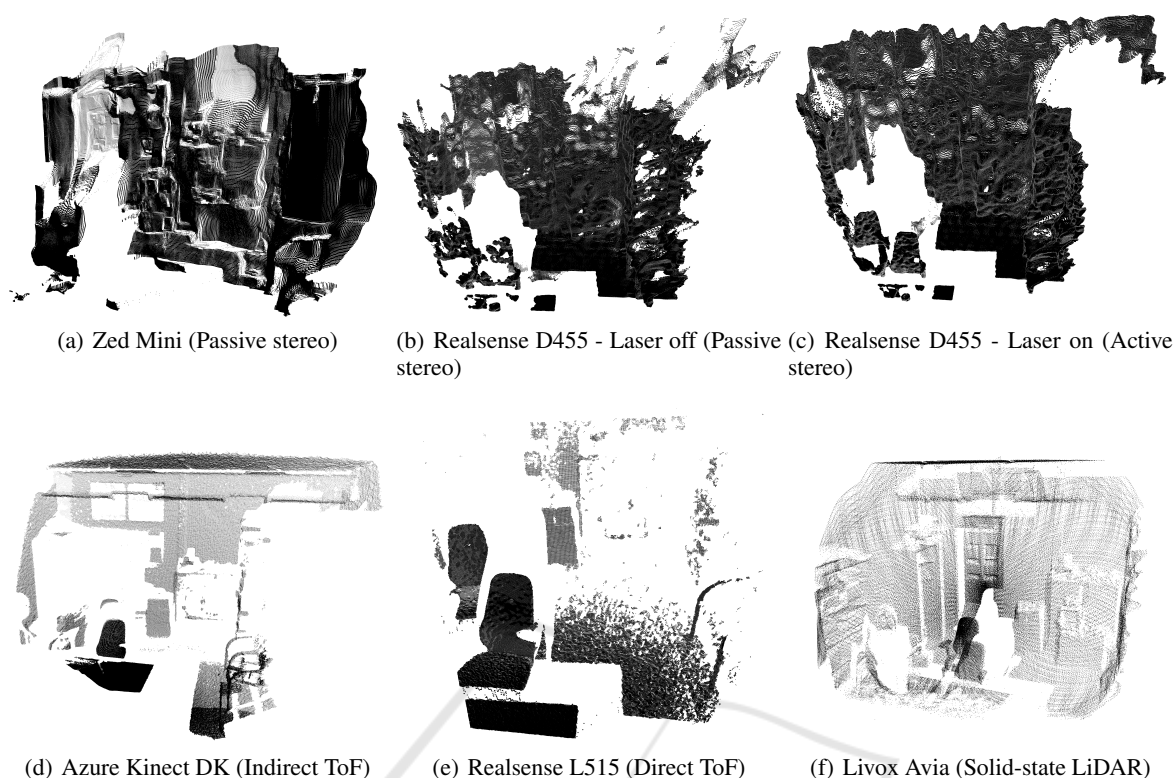


Figure 4: Point clouds obtained with various depth cameras and with the Livox Avia solid-state LiDAR for reference.

or techniques will allow us to draw high-level insights on where these methods still fail to produce accurate results. We also capture a reference point cloud of the scene with a Livox Avia solid-state LiDAR (Livox, 2021).

We start with the evaluation of depth cameras on our sample scene. All depth cameras are used with default settings, except the Intel Realsense D455 with which we capture the scene with the laser off (passive stereo) and on (active stereo). The resulting point clouds are shown in Figure 4. On that figure, we also put the point cloud acquired with a Livox Avia solid-

state LiDAR for reference. Note that different sensors have different fields of view (FoV), which also affects the general outlook of the point cloud.

- The Zed Mini (passive stereo) shows a very dense but distorted point cloud, with an abnormal structure appearing far away above the lab door. The distortion is expected, since the stereo matching cannot be reliable in several areas of an indoor setting with flat or texture-less surfaces. Then, the structure above the lab door can be explained since a stereo vision-only sensor cannot find features to match and triangulate in a glass window with the sky behind.
- The Realsense D455, in passive stereo mode, shows much higher variance, and wave-like ghost structures appear in the whole depth map.
- The Realsense D455, in active stereo mode, shows increased accuracy in nearby structures (table and chairs), but the wave-like structures remain present in the structures a few meters away (closet, wall and door).
- The Azure Kinect DK, with its ToF sensor, has a much wider field of view, an interesting property for 3D reconstruction. It outputs slightly fewer points, but exhibits low bias and variance: flat



Figure 5: Our test scene. It features cluttered areas, planar surfaces, various materials and various reflections, parts with external lighting (through a window), and parts with poor lighting.


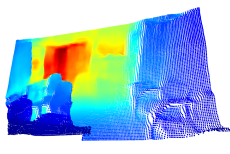
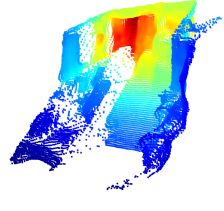

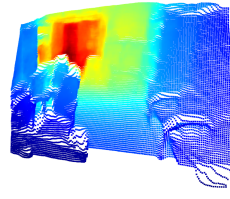
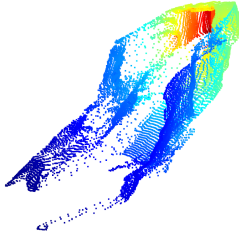

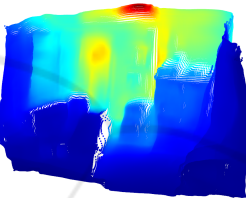
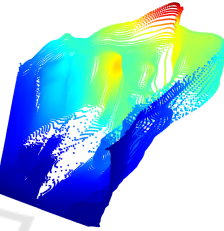

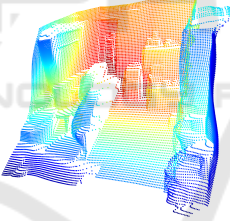
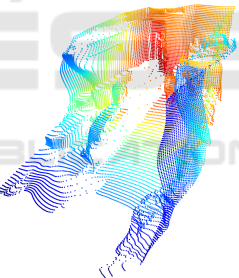
Algorithm	Depth map	Close point of view	Distant point of view
DepthFormer			
BinsFormer			
TCMonoDepth			
SC-Depth			

Figure 6: Depth maps returned from the depth estimation algorithms, and reconstructed point clouds seen from two different points of view. The coloring is relative to the position of the point on the forward axis.

structures are flat, with relatively small noise.

- The Realsense L515, using MEMS LiDAR, shows good accuracy on nearby structures (table and chairs), but this degrades with more distant ones (wall, ground). However, the noise remains lower than with stereo-vision sensors.

All of these results are very different. Now, it is expected that stereo vision does not perform too well in indoor settings, due to the lack of texture, relative to outdoor environments. The depth cameras using other modalities output better results on our test scenario. Then, the ToF camera (Azure Kinect) shows more reliable points. However, in this test, these two systems also suffer from a decreased depth map density compared to the others, as shown in the following table:

Table 1.

Depth camera	Resolution	Density
Zed Mini	1920 × 1080	96%
RS D455 - laser off	848 × 480	74%
RS D455 - laser on	848 × 480	90%
MS Azure Kinect DK	1024 × 1024	47%
RS L515	640 × 480	48%

We now move on to the deep learning-based dense depth estimation algorithms. Since we are using an indoor image for our tests, we use weights resulting from training on the NYU Depth dataset. We feed the same RGB image to DepthFormer, BinsFormer, TCMonoDepth and SC-Depth. The results are shown in Figure 6, where we display the depth maps and the

point clouds computed using the de-projection operator  $\pi_{\mathbf{K}}^{-1}$ . We use the intrinsic matrix  $\mathbf{K}$  associated to the camera with which the RGB image was taken. All depth maps use the same “magma” color mapping, although the absolute scale is estimated by each algorithm. We display the point clouds from two points of view, one close to the initial location of the camera and one from a distant point of view. We do this because, since the algorithms return fully dense depth maps, it is hard to visualize the geometry of the point cloud from the initial point of view. Again, point clouds may appear to have different scales depending on the depth map resolution and the estimated scale. Additionally, we color the point clouds with a color map relative to the position of the points on the forward axis to better distinguish the different objects.

- The depth map from DepthFormer shows a slight lack of detail in some structures, leading to ghost connections (e.g. between the arms of the chairs). Looking at the point cloud from a close point of view does not reveal many errors besides an exaggerated depth map smoothness (making unconnected objects appear connected) and slight distortions. On the other hand, the distant point of view highlights the heavy distortions in the geometry of the wall and the closet.
- BinsFormer has a performance close to DepthFormer, if we look at the depth map. The thin structures appear slightly more detailed. However, we see with the distant viewpoint that the scale is more wrong.
- TCMonoDepth has fewer details and the same error with the window above the lab door; it is estimated to be far away. Another important error of this model is also linked to exaggerated smoothness: looking at the distant point of view, the walls and closets appear very rounded.
- SC-Depth shows a great level of detail in the depth map with very few wrong connections or ghost structures. Albeit better than the previous models, the walls and closets still look somewhat rounded and distorted.

All of these methods produce relatively high resolution depth maps and contrarily to the depth cameras, depth estimation neural networks output depth maps without any hole. The scene can be recognized in all depth maps, but not with the same level of detail. Arguably the most important issue is the exaggerated smoothness: the whole point cloud appears as a connected surface, lacking details (e.g., the void between the chair arms is filled in three out of four point clouds). Angles and surfaces are also severely altered with all deep models. Finally, examining the

point clouds as seen from a more distant point of view, we notice that the scale of these depth maps can be wrong. But, to be fair, this is expected, as there is no way an algorithm using only monocular images could compute the absolute scale. Although not an error of the algorithm, but rather a fundamental limitation, this adds to the list of challenges to solve before using these depth maps in real applications.

## 5 CONCLUSION

Let us start with the remark that both categories of methods have clear strengths and shortcomings. None of the propositions is really a definite go-to, one-size-fits-all method. Although our evaluation only considers a single image, fundamental characteristics appear to be common for all depth cameras or depth estimation algorithms: depth cameras with ToF and MEMS LiDAR technology provide accurate geometry, but have relatively fewer points. Depth estimation algorithms rather suffer from geometry issues such as exaggerated smoothness and distorted structures, but output fully dense depth maps. In the context of 3D reconstruction with portable systems, depth cameras with ToF or MEMS LiDAR are, for now, more adequate: despite the lower number of points, point cloud registration can still be achieved as the errors in depth sensing remain mostly centered around zero. Hence the abundant literature on 3D reconstruction with such sensors. Point cloud registration with the geometrically-inaccurate clouds from deep depth maps, on the other hand, is extremely challenging: all points are very well grouped (low variance), but not necessarily in the right place (high bias), which makes the registration and fusion extremely difficult. Considering the above observations, an interesting research direction would be to fuse depth maps from depth sensors with deep learning depth estimation methods, i.e. performing depth densification.

## ACKNOWLEDGMENTS

This work is part of the Scientific and Technological Research of Defence Program of Belgium, and received financial support by the Royal Higher Institute for Defence under project name DAP18/04.

## REFERENCES

- Bae, G., Budvytis, I., and Cipolla, R. (2022). Multi-view depth estimation by fusing single-view depth proba-

- bility with multi-view geometry. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Dong, X., Garratt, M. A., Anavatti, S. G., and Abbass, H. A. (2021). Towards real-time monocular depth estimation for robotics: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 23:16940–16961.
- Geiger, A., Lenz, P., and Urtasun, R. (2012). Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Guizilini, V., Ambrus, R., Chen, D., Zakharov, S., and Gaidon, A. (2022). Multi-frame self-supervised depth with transformers. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Hartley, R. and Zisserman, A. (2003). *Multiple View Geometry in Computer Vision*. Cambridge University Press, New York, NY, USA, 2 edition.
- Intel (2020a). Intel Realsense D455. <https://www.intelrealsense.com/depth-camera-d455/>. [Online; accessed 20-November-2022].
- Intel (2020b). Intel Realsense L515. <https://www.intelrealsense.com/lidar-camera-l515/>. [Online; accessed 20-November-2022].
- Janoch, A. (2012). The berkeley 3d object dataset. Master’s thesis, EECS Department, University of California, Berkeley.
- Li, S., Luo, Y., Zhu, Y., Zhao, X., Li, Y., and Shan, Y. (2021). Enforcing temporal consistency in video depth estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*.
- Li, Z., Wang, X., Liu, X., and Jiang, J. (2022). Binsformer: Revisiting adaptive bins for monocular depth estimation. *arXiv preprint arXiv:2204.00987*.
- Livox (2021). Livox Avia. <https://www.livoxtech.com/avia>. [Online; accessed 20-November-2022].
- Microsoft (2020). Microsoft Azure Kinect DK. <https://azure.microsoft.com/en-us/products/kinect-dk/>. [Online; accessed 20-November-2022].
- Ming, Y., Meng, X., Fan, C., and Yu, H. (2021). Deep learning for monocular depth estimation: A review. *Neurocomputing*, 438:14–33.
- Nathan Silberman, Derek Hoiem, P. K. and Fergus, R. (2012). Indoor segmentation and support inference from rgb-d images. In *ECCV*.
- Song, S., Lichtenberg, S. P., and Xiao, J. (2015). Sun rgb-d: A rgb-d scene understanding benchmark suite. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 567–576.
- Sun, L., Bian, J.-W., Zhan, H., Yin, W., Reid, I., and Shen, C. (2022). Sc-depthv3: Robust self-supervised monocular depth estimation for dynamic scenes. *arXiv:2211.03660*.
- Tychola, K., Tsimperidis, I., and Papakostas, G. (2022). On 3d reconstruction using rgb-d cameras. *Digital*, 2:401–423.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. (2017). Attention is all you need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.
- Wang, H., Wang, C., and Xie, L. (2021). Lightweight 3-d localization and mapping for solid-state lidar. *IEEE Robotics and Automation Letters*, 6(2):1801–1807.
- Xiao, J., Owens, A., and Torralba, A. (2013). Sun3d: A database of big spaces reconstructed using sfm and object labels. *2013 IEEE International Conference on Computer Vision*, pages 1625–1632.
- Yuan, C., Xu, W., Liu, X., Hong, X., and Zhang, F. (2022a). Efficient and probabilistic adaptive voxel mapping for accurate online lidar odometry. *IEEE Robotics and Automation Letters*, 7:8518–8525.
- Yuan, W., Gu, X., Dai, Z., Zhu, S., and Tan, P. (2022b). Newcrfs: Neural window fully-connected crfs for monocular depth estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.
- ZED (2017). ZED Mini . <https://www.stereolabs.com/zed-mini/>. [Online; accessed 20-November-2022].
- Zhang, S., Zheng, L., and Tao, W. (2021). Survey and evaluation of rgb-d slam. *IEEE Access*, 9:21367–21387.
- Zollhöfer, M., Stotko, P., Görnitz, A., Theobalt, C., Nießner, M., Klein, R., and Kolb, A. (2018). State of the art on 3d reconstruction with rgb-d cameras. *Computer Graphics Forum*, 37.