Revista de Informática Teórica e Aplicada - RITA - ISSN 2175-2745
Vol. 32, Num. 1 (2025) 166-172

RESEARCH ARTICLE

Case Study of Deep Learning Methods for Depth Estimation in Indoor Ground Robotics

Estudo de Caso de Métodos de Aprendizado Profundo para Estimação de Profundidade em Robótica Terrestre de Interior de Ambiente

Fábio Leandro Vizzotto[1], Marcos D'Addio de Moura[1], Vinicius Carbonezi de Souza[1]*, Cides Semprebom Bezerra[1], Guilherme Ribeiro Sales[1], Valentino Corso[1], Luiz Eduardo Pita Mercês Almeida[1], Douglas Henrique Siqueira Abreu[2]

Abstract: Depth estimation is the computer vision task that assigns a distance between the camera and each pixel in an image. This paper focuses on monocular metric depth estimation in videos, which infers a distance in metric units using a single RGB camera. Considering its applications, robotics systems and environmental mapping arise as practical areas that can make extensive usage of these techniques. As a case study for indoor robotics, the ICL ground robot dataset obtained by video footage in graphic simulation was used for experiments. A comparison was made considering the results and requirements of data acquisition needed for different deep learning models, presenting self-supervised and supervised methods available in literature and being the first work to present a depth estimation benchmark for the chosen dataset.
Keywords: Depth Estimation — Ground Robotics — Deep Learning — Case Study

Resumo: A estimação de profundidade é a tarefa de visão computacional que atribui uma distância entre a câmera e cada pixel em uma imagem. Este trabalho se concentra na estimação de profundidade métrica monocular em vídeos, a qual infere uma distância em unidades métricas usando uma única câmera RGB. Considerando suas aplicações, sistemas de robótica e mapeamento ambiental são as áreas práticas que podem fazer uso extensivo dessas técnicas. Como um estudo de caso para robótica interna, o conjunto de dados *ICL ground robot* obtido por filmagem de vídeo em simulação gráfica foi usado para os experimentos. Uma comparação foi feita considerando os resultados e requisitos de aquisição de dados necessários para diferentes modelos de aprendizado profundo, apresentando métodos autossupervisionados e supervisionados disponíveis na literatura e sendo o primeiro trabalho a apresentar resultados métricos de estimação de profundidade para o conjunto de dados escolhido.
Palavras-Chave: Estimação de Profundidade — Robótica Terrestre — Aprendizado Profundo — Estudo de Caso

[1] *Artificial Intelligence and IoT Solutions, Centro de Pesquisa e Desenvolvimento em Telecomunicações (CPQD), Brazil*
[2] *Computer Science Department, Pontifícia Universidade Católica de Campinas (PUCC), Brazil*
***Corresponding author**: viniciusc@cpqd.com.br
DOI: http://dx.doi.org/10.22456/2175-2745.143443 • **Received:** 22/10/2024 • **Accepted:** 23/12/2024
CC BY-NC-ND 4.0 - This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License.

1. Introduction

Depth estimation using computer vision is a research area focused on the inference of the distance between the camera and the objects captured in an image, adding a new dimension to the understanding of the data [1].

The accomplishment of this task is fundamental for a range of critical technological downstream tasks to be executed, such as autonomous navigation and three-dimensional environmental mapping. By accurately interpreting depth from images, smarter and more adaptable systems can be developed in the dynamic fields of robotics and augmented reality, for example.

In the absence of computer vision methods, depth measurement can be conducted using specific sensors, such as LiDARs and RGB-D cameras. However, depending on the LiDAR system technology used, the depth map generated can present a spatially sparse distribution, complicating the localization of smaller objects and edges [2]. On the other hand, RGB-D cameras often operate within a limited optimal

operation depth range [3].

These exposed negative characteristics of depth sensors, along with the need for frequent calibration procedures, underscores the importance of exploring robust methods as viable alternatives for depth acquisition. In this context, deep learning approaches are used to overcome these shortcomings [4, 5, 6, 7, 8].

In this context, the input considered for the depth estimation method can be a single or a set of images, which determine the number of cameras used in the hardware setup and can be named as monocular, stereo or multi-view. Monocular methods use a single image as input, thus requiring only one camera. The more traditional stereo methods use two cameras, similar to human vision. Finally, multi-view methods use more than two cameras [4].

As monocular methods use a single camera, this approaches are preferred in situations that need reduced mount space, minimized costs and little re-calibration procedures, both for the camera itself and for the distances in the physical setup [9]. Therefore, monocular depth estimation approaches have suitable practical characteristics to be applied in autonomous robotics systems, with the downside of being more challenging as an intrinsic mathematical ill-posed problem [4], being the focus of this paper.

Another division of depth estimation approaches rises from the output produced by the method, which can be classified as a relative or metric depth [7].

Relative depth methods assign minimum and maximum values to the nearest and farthest objects, respectively. As a consequence of this normalization characteristic, generic depth estimators usually are relative [5, 6], fairly training the multiple scene scales of diverse datasets. However, its practical applications can be reduced due to the scene specific post-processing techniques required to obtain real physic estimations [10].

On the other hand, metric depth methods infer values in consistent real physical metric units, such as meters, generally with limited bounds for estimation. These methods are generally present in specialized solutions for scenes similar to the ones used as training [7, 8]. In this way, they present lesser generalization capabilities in the wild, but are ready to use in tasks like mapping and navigation.

In this paper, monocular metric depth estimation methods are compared considering specifications of data necessity and performance in regression errors and thresholded accuracy. The ICL dataset [11] was used for the case study, comprehending video footage from a graphic simulation of a ground indoor robot, with experiments proposed and results obtained considering its practical usage.

Being the first authors to explore the ICL ground robot dataset for depth estimation, it is expected to stimulate further development and research involving this public resource, which contains dense and high quality depth maps for supervised learning methods, as well as time continuity over frames, enabling the usage of popular self-supervised learning geometry-based techniques.

The main contributions of this work are: (i) Provide a reference case study for depth estimation in indoor ground robotics; (ii) Present the first results of depth estimation for the public ICL ground robot dataset; (iii) Perform a comparison between different self-supervised and supervised deep learning depth estimation methods in the dataset.

This work is divided as follows: Section 2 presents literature related works on computer vision depth estimation, Section 3 describes the dataset and metrics used in experiments, Section 4 describes the experiments done and presents quantitative and qualitative results obtained and Section 5 summarizes the observed conclusions.

2. Related Works

Classical computer vision depth estimation methods generally acquire images and compare results from different frames using epipolar geometry equations, in which the results are obtained using closed form solution formulas [8].

Although these approaches analytically solves the depth estimation problem, the usage of variables such as the intrinsic characteristics of the cameras and distances in the mount setup have a direct impact in the accuracy generated, which can lead to cumulative errors caused by inadequate or absent calibration procedures [4]. Other practical limitation aspects are the assumption of environmental spatial constraints or camera trajectory, which difficult its operation in non-controlled scenarios [1].

Nowadays, in order to address these characteristic issues, deep learning methods are employed to provide more general and robust solutions.

2.1 Self-supervised Methods

Self-supervised methods for depth estimation focus on solving the problems generated by the necessity of acquisition of large amounts of high-quality dense data for training [4].

Widely explored, the Monodepth2 [8] solution is trained using a collection of past frames in a monocular video to recreate the current frame by using a pose estimator network and a depth projection network, which is the main focus of training. Photometric loss is used to evaluate the difference between the current frame predicted and the actual recorded in the dataset, with optical flow analyzed to avoid high penalties due to object occlusion.

Using similar self-supervised training methods along with distillation techniques, the DistDepth [10] approach uses an expert generalist model combined with data specific self-supervised learning to train a less complex network and generate metric depth.

2.2 Supervised Methods

Predominant in current works for supervised depth estimation, the DPT architecture [12] uses transformer-based encoders to accomplish dense prediction tasks such as segmentation. Its

authors reported that this architecture was capable of generating models that outperformed previous state of art approaches in depth estimation, accomplishing it by using the pre-trained model and applying fine tune in specific depth estimation datasets.

In this context, new solutions were built using the DPT architecture aiming to create a generic depth estimator to be used as a foundation model. These methods usually generate relative depth, normalizing distances of the ground truth to address fairly training between different datasets scene scales. As an example, this strategy is used by MiDaS [5], trained in five different indoor and outdoor datasets.

Another general depth estimator example is Depth Anything [6], which used unlabeled images during training with pseudo labels generated by other state of art depth estimation methods. This way, the data used for training was bigger than previous solutions, leading to best results.

As a way to use the generalists models to specific applications, ZoeDepth [7] internally uses a pre-trained relative depth estimator model and converts its output to a metric unit, allowing the direct usage of the result in real world scenes after a supervised fine tune training. This is done by using a convolutional head called metric bins module, which is combined with the generalist decoder to obtain the final depth output by aggregating the intermediate relative depth output.

2.3 Graphical Simulation

Graphical simulation of real world similar scenes are often used in the depth estimation training as an alternative to the acquisition of real world data [10]. In this manner, the simulated environment presents the benefit to facilitate high-quality dense ground truth attainment and generation of larger datasets compared to classical sensor based acquisited datasets.

In regard of the performance of the resultant models in real data, the authors in [10] present a comparison between a deep learning model trained only with simulated data and a model uniquely trained with real world data, achieving a performance difference lesser than 5% in regression errors and depth estimation accuracy when evaluated in the NYUv2 dataset [3]. Thus reinforcing the validity of the usage of simulation to accelerate the development of depth estimation techniques, as it was done in this paper.

3. Materials and Methods

This section describes the dataset and quantitative metrics used throughout the experiments in this work.

3.1 Dataset

This work uses the ICL ground robot dataset [11], in which the movement trajectory is based on collected sensor data from a real robot and the realistic environment scenes are graphically generated. Samples of the scenes contained in the dataset are presented in Figure 1.

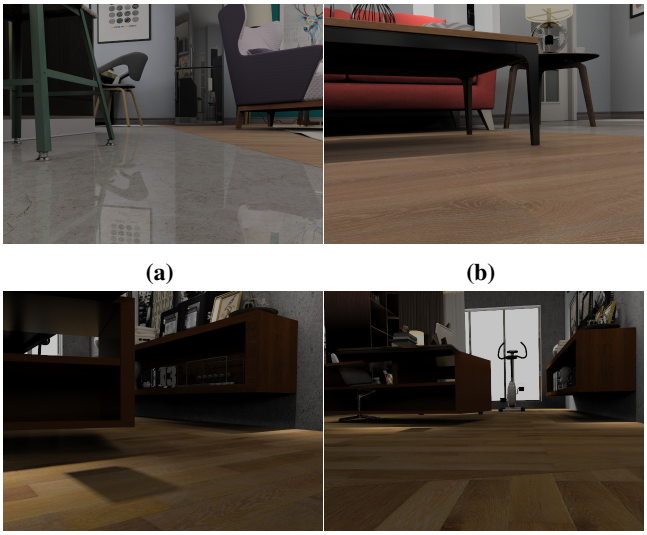


Figure 1. Samples of the dataset used. (a) and (b): Deer scene. (c) and (d): Diamond scene. Adapted from [11].

The dataset is composed of two residential indoor scenes, named Deer and Diamond, differing in light illumination, style and positioning of typical household objects. Each scene is composed of 1600 monocular frames, collected at 20 FPS.

Differently from other public datasets for depth estimation, such as KITTI [2] and NYUv2 [3], the ICL dataset [11] provides a continuous video footage with dense generated ground truth. In this way, it facilitates the usage of different techniques for depth estimation. It is possible to apply methods that rely on smooth camera movement, in other cases hampered in datasets without scene continuity [8] and also makes direct the application of supervised training methods without the need of depth completion to enhance data quality for reasonable results [7].

In this context, the ICL dataset was chosen due the characteristic to enable a broad range of methods to be compared, along with its photo-realistic characteristic, exploring its potential as benchmark for diverse depth estimation methods.

As far as our knowledge, there is no indication of suggested data splitting by the ICL dataset authors [11] or in other works for the depth estimation task. Therefore, in this paper the dataset is split by using the first 70% of the frames for training, the subsequent 10% for validation and the last 20% for test, applying this logic for each scene.

For the results calculations and comparisons of performance, the range of minimum and maximum metric distances considered for the ground truth were 1 *mm* to 10 *m* relative to the camera.

3.2 Metrics

To enable quantitative comparison between experiments, the regression errors analyzed were mean absolute relative error (*Abs Rel*) presented by Eq. 1, mean squared relative error (*Sq Rel*) in Eq. 2, root mean squared error (*RMSE*) in Eq. 3

and root mean squared logarithmic error (*RMSE log*) in Eq. 4. The calculations formulas are described in [9].

$$\text{Abs Rel} = \frac{1}{|N|}\sum_{i \in N}\frac{|d_i - d_i^*|}{d_i^*} \tag{1}$$

$$\text{Sq Rel} = \frac{1}{|N|}\sum_{i \in N}\frac{|d_i - d_i^*|^2}{d_i^*} \tag{2}$$

$$\text{RMSE} = \sqrt{\frac{1}{|N|}\sum_{i \in N}|d_i - d_i^*|^2} \tag{3}$$

$$\text{RMSE log} = \sqrt{\frac{1}{|N|}\sum_{i \in N}|\log(d_i) - \log(d_i^*)|^2} \tag{4}$$

The thresholded accuracies observed were $\delta < 1.25$, $\delta < 1.25^2$ and $\delta < 1.25^3$, presented in Eq. 5, as demonstrated in [9].

$$\delta < \text{thr} : \%\text{ of } d_i \text{ s.t. } \max\left(\frac{d_i}{d_i^*}, \frac{d_i^*}{d_i}\right) < \text{thr} \tag{5}$$

In which d_i is the depth estimated and d_i^* is the respective ground truth for pixel i. The metrics are calculated considering N as the total number of pixels with valid ground truth.

For experiment comparison, the most significant metric of regression error chosen was the *RMSE log*, due to its normalization characteristics for different distances. For accuracy, the $\delta < 1.25$ was given more importance, as it demonstrates the fine accuracy of the model to estimate the depth [9].

4. Experiments and Results

Experiments were conducted using the ICL ground robot dataset subdivided as explained in Section 3.1, each model was trained based on the open source implementation of the network architecture distributed by the original authors with adapters to correctly load the new dataset. For the same approach, the best model was chosen by varying hyperparameters and selecting the experiment that returned the best metrics in the validation subset accordingly to the preferences mentioned in Section 3.2.

Qualitative analysis of the depth maps generated were also used to propose and guide parameters value changes.

4.1 Supervised model

The experiments used the MiDaS variation named *dpt swin2-T* network, a DPT model with a swin transformer v2 [13] backbone considered by its authors as having a balanced trade-off between accuracy and hardware consumption.

Qualitative results for the relative depth generated in the dataset presented promising edge depth estimation and object differentiation in off the shelf usage. Therefore, this variation was chosen to be the generalist model for further experiments with ZoeDepth for metric depth estimation.

For the ZoeDepth experiments, the variation named *DPT SwinV 2 T 256* by the original paper was chosen, referring to the generalist model MiDaS *dpt swin2-T* with an input image size of 256x256 pixels. The metric depth head of the models experimented was originally pre-trained with the NYUv2 [3] dataset by its authors, suffering fine tune to the ICL ground robot dataset in each training.

All the experiments described were done using horizontal flip data augmentation, with an AdamW [14] optimizer and a OneCycleLR [15] scheduler for 20 epochs.

The results obtained in experiments that adopted a non-trainable (frozen) depth generalist network approach in which the training only adjusted the metric heads weights are presented in the Table 1, in which all experiments were done with the default starting learning rate of 10^{-4}.

Considering the experiments in the Table 1, the number of bins in the metric head was explored from the standard value of 64, the best found by the original authors [7], being reduced to 32 and 16. The reduction in this parameter was proposed due to the qualitative presence of nonexistent depth artifacts for the default value in the dataset, with lower values tending to cause depth grouping for different objects, thus dealing with the imperfections observed.

From the Table 1, the usage of 32 metric bins in the head (experiment *Frozen-32*) was considered to yield the best results. As prominent values, the *RMSE log* of 0.237 and the accuracy $\delta < 1.25$ of 0.736 demonstrate the superiority of this experiment over the others.

Although relative similar regression error values are presented by the 16 bins experiment (*Frozen-16*), lower fine accuracy ($\delta < 1.25$) results were achieved due to depth grouping, which oversimplified the metric depth generated.

Considered the best candidate from the Table 1, the reduced number of 32 bins in the metric module was used throughout the experiments conducted for the ZoeDepth models with trainable internal generalist depth estimator. In these experiments a conjunction training were done, fine tuning the generalist network along with the metric depth heads. The results obtained are presented in Table 2.

For the Table 2, the results obtained demonstrated that the usage of a reduced starting learning rate of 10^{-5} in experiment *Trainable-5* returned the best metrics, presenting *RMSE log* of 0.144 and $\delta < 1.25$ of 0.850.

Comparing the results of the generalist non-trainable and trainable approaches in Tables 1 and 2, the trainable variations presented overall better results. With this, the fine tune of the generalist model demonstrated to be effective to achieve better metrics, despite its higher complexity of training.

Finally, the trainable method using the reduced 32 bins in metric head and with reduced learning rate (experiment *Trainable-5*) was chosen as the best ZoeDepth model obtained. The results achieved after training this best model setup for

Table 1. ZoeDepth Frozen Generalist Experiments Results in Validation Subset

Identification	Metric bins	Abs Rel	Sq Rel	RMSE (m)	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Frozen-64	64	0.226	0.154	**0.474**	0.243	0.699	0.879	0.957
Frozen-32	32	**0.218**	0.151	0.480	**0.237**	**0.736**	**0.890**	0.953
Frozen-16	16	0.220	**0.150**	0.479	0.239	0.715	0.883	**0.959**

Table 2. ZoeDepth Trainable Generalist Experiments Results in Validation Subset

Identification	Learning Rate	Abs Rel	Sq Rel	RMSE (m)	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Trainable-4	10^{-4}	0.146	0.062	**0.286**	0.159	0.836	**0.937**	0.991
Trainable-5	10^{-5}	**0.121**	**0.051**	0.288	**0.144**	**0.850**	0.871	**0.996**

40 epochs are presented in Table 3 for the test subset.

4.2 Self-supervised model

Experiments with the Monodepth2 were done by fine tuning of the monocular model variation *mono_640x192*, pre-trained on the KITTI [2] dataset. The depth estimator model presents an U-Net [16] architecture with a ResNet-18 [17] backbone.

As specific monocular training conditions experimented, it was chosen the usage of a separated ResNet-18 pose estimator network, aiming for independence of weights between the pose and the depth estimators.

For the frame generation algorithm, focus of the self-supervised training, it was used two consecutive past frames as input for the depth and pose estimators networks to produce the estimated present frame.

Training was accomplished using an Adam [18] optimizer with a stepLR scheduler for 20 epochs and reduced smooth factor of 10^{-5}, horizontal flipping data augmentation was applied.

Results for the test subset using the best Monodepth2 model trained are presented in Table 3.

4.3 Comparison

A qualitative comparison of the models in Table 3 is presented in Figure 2.

According to the qualitative results of the depth maps generated, the ZoeDepth model estimates depth maps with more defined object borders, as depicted in Figure 2b, being a more suitable solution for clustered scenes navigation and mapping.

The Monodepth2 trained model results demonstrated in Figure 2c presented a tendency of generating a blurred output, limiting its practical usage for applications that do not need information on edge detection or fine distinction of objects.

The comparison of quantitative metrics in Table 3 demonstrates the better results of the ZoeDepth trained model for all metrics considered, except by the *Sq Rel* and *RMSE* regression error metrics, in which the Monodepth2 has lower values.

These differences in the metric results behavior can possibly be explained by the fact that ZoeDepth presented an edge transition sharper and better defined than the Monodepth2 smooth output. As mathematically exists a major penalty of the quadratic errors by the calculation formulas in Eq. 2 and Eq. 3, a much higher numerical value for the pixel error is generated when an object is misplaced by the background or when the inverse situation occurs, generally on object edges.

In this context, as there is no normalization in these metrics formulas, higher individual pixel errors are generated for clear defined slightly misplaced objects as in ZoeDepth, situation in which a quadratic penalty is applied to object-background confusion. In comparison, blurred mean depth map outputs, similar to Monodepth2, can present lower pixel errors due to the numerically inferior quadratic penalty for the confusion between the mean distance estimated and the background. As the mean of errors for all pixels in the depth map is calculated in the final metric, the high penalty for object-background confusion in ZoeDepth numerically elevate its mean error more than the common mean depth and background (or mean depth and foreground object) error for Monodepth2.

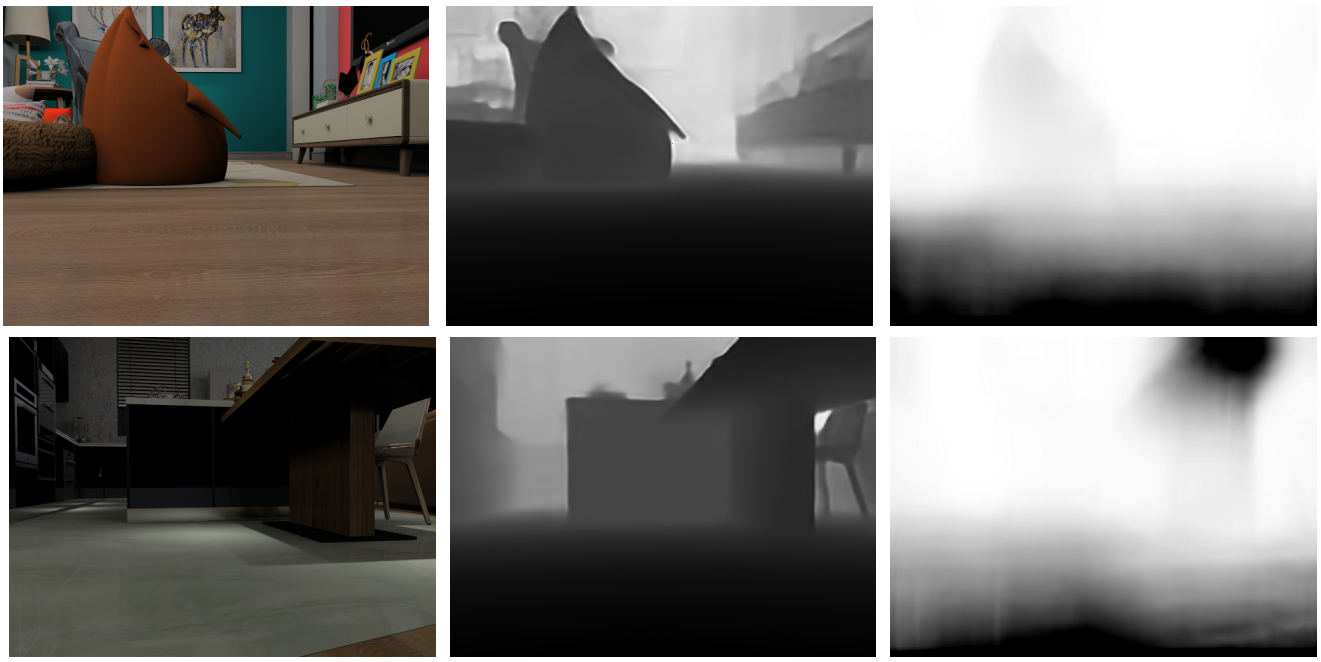
On the other hand, the computational complexity of the Monodepth2 model is significantly lower when compared with the ZoeDepth. The major aspect contributing to this is the usage of the more complex generalist network during inference time for the ZoeDepth, which can be a limitation for embedded robotic systems considering its greater time required for processing.

Another advantage of the self-supervised model is the characteristic of the training process occurring independently from the ground truth collected. Being of interest for depth estimation as real world sensors can generate sparse data, which pose difficulties for supervised training in dense prediction tasks [6]. However, recent works relying on computer graphics or generative AI for the dataset acquisition tend to reduce this advantage by providing dense depth maps [10].

As there is no other prior works on depth estimation using the ICL ground robot dataset in literature, a comparison with our results and other state-of-the-art methods should be done with caveats. The metrics presented numerically by the original authors of ZoeDepth [7] and Monodepth2 [8] trained in classic sparse depth estimation datasets such as KITTI [2] and NYUv2 [3] suggests that could exist room for improvement

Table 3. Comparison of Model Results

Model	Abs Rel	Sq Rel	RMSE (m)	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
ZoeDepth	**0.258**	0.151	0.651	**0.276**	**0.658**	**0.870**	**0.949**
Monodepth2	0.404	**0.099**	**0.162**	0.445	0.515	0.787	0.882



(a) **(b)** **(c)**

Figure 2. Qualitative comparison of depth maps generated by the models. (a) Input image. (b) ZoeDepth output. (c) Monodepth2 output.

for our results based on the metrics produced. However, for a fair comparison, the difference in depth map ground truths characteristics, dense or sparse, must be considered.

5. Conclusion

This work presented a comparative case study for monocular metric depth estimation in the context of indoor robotics. To accomplish it, different depth estimation methods were applied to the until now unexplored ICL ground robot dataset, with realistic simulation scenes that can be further used by downstream tasks or have its benchmark expanded for new depth estimation methods, given its generality and capacity to be related to the real world scenes.

The self-supervised Monodepth2 model was presented as a candidate to accomplish the task, with lower dependency on high quality ground truth acquired data, also demanding lower computational complexity for inference, but at the cost of coarse generated depth maps, difficulting its application for precise navigation and mapping, as observed in Figure 2c.

In contrast, the supervised ZoeDepth model results obtained overall better quantitative metrics in the dataset used, except for quadratic penalty metrics without distance normalization. Qualitatively was able to differ smaller objects, as depicted in Figure 2b, thus considered more suitable to accurate tasks. However, its downsides include the increased dependency of annotated data and increased hardware requirements for inference.

Acknowledgements

This project was supported by Ministério da Ciência, Tecnologia e Inovações, with funding from law no 8.248, of October 23, 1991, within the scope of PPI-SOFTEX, coordinated by Softex and published PDI 03, DOU 1245.023862/2022-14.

Author contributions

Fábio Leandro Vizzotto, Marcos D'Addio de Moura and Vinicius Carbonezi de Souza: main authors, performed and suggested experiments in model training, analysed metric results; Cides Semprebom Bezerra, Guilherme Ribeiro Sales, Valentino Corso: secondary authors, suggested experiments in model training; Luiz Eduardo Pita Mercês Almeida: co-advisor, suggested experiments in model training, guided article writing; Douglas Henrique Siqueira Abreu:Advisor, guided and suggested experiments in model training.

References

[1] LUO, X. et al. Consistent video depth estimation. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH)*, ACM, v. 39, n. 4, 2020.

[2] GEIGER, A.; LENZ, P.; URTASUN, R. Are we ready for autonomous driving? the kitti vision benchmark suite. In: IEEE. *2012 IEEE conference on computer vision and pattern recognition*. [S.l.], 2012. p. 3354–3361.

[3] SILBERMAN, N. et al. Indoor segmentation and support inference from rgbd images. In: SPRINGER. *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part V 12*. [S.l.], 2012. p. 746–760.

[4] SPENCER, J. et al. The monocular depth estimation challenge. In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. [S.l.: s.n.], 2023. p. 623–632.

[5] RANFTL, R. et al. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE transactions on pattern analysis and machine intelligence*, IEEE, v. 44, n. 3, p. 1623–1637, 2020.

[6] YANG, L. et al. Depth anything: Unleashing the power of large-scale unlabeled data. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. [S.l.: s.n.], 2024. p. 10371–10381.

[7] BHAT, S. F. et al. Zoedepth: Zero-shot transfer by combining relative and metric depth. *arXiv preprint arXiv:2302.12288*, 2023.

[8] GODARD, C. et al. Digging into self-supervised monocular depth estimation. In: *Proceedings of the IEEE/CVF international conference on computer vision*. [S.l.: s.n.], 2019. p. 3828–3838.

[9] ZHAO, C. et al. Monocular depth estimation based on deep learning: An overview. *Science China Technological Sciences*, Springer Science and Business Media LLC, v. 63, n. 9, p. 1612–1627, jun. 2020. ISSN 1869-1900. Disponível em: ⟨http://dx.doi.org/10.1007/s11431-020-1582-8⟩.

[10] WU, C.-Y. et al. Toward practical monocular indoor depth estimation. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. [S.l.: s.n.], 2022. p. 3814–3824.

[11] SAEEDI, S. et al. Characterizing visual localization and mapping datasets. In: IEEE. *2019 International Conference on Robotics and Automation (ICRA)*. [S.l.], 2019. p. 6699–6705.

[12] RANFTL, R. et al. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2020.

[13] LIU, Z. et al. Swin transformer v2: Scaling up capacity and resolution. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. [S.l.: s.n.], 2022. p. 12009–12019.

[14] LOSHCHILOV, I. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[15] SMITH, L. N.; TOPIN, N. Super-convergence: Very fast training of neural networks using large learning rates. In: SPIE. *Artificial intelligence and machine learning for multi-domain operations applications*. [S.l.], 2019. v. 11006, p. 369–386.

[16] RONNEBERGER, O.; FISCHER, P.; BROX, T. U-net: Convolutional networks for biomedical image segmentation. In: SPRINGER. *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*. [S.l.], 2015. p. 234–241.

[17] HE, K. et al. Deep residual learning for image recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.l.: s.n.], 2016. p. 770–778.

[18] KINGMA, D. P. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.