Depth Learner : An Advanced Learning Method with Edge Map for Monodepth2

Minseung Kim, Seongmin Kim, Junmyeong Kim, Kanghyun Jo
Dept. of Electrical, Electronic and Computer Engineering
University of Ulsan, Ulsan, Korea
{kmsoiio, asdfhdsa1234}@mail.ulsan.ac.kr, kjm7029@islab.ulsan.ac.kr, acejo@ulsan.ac.kr

2023 International Workshop on Intelligent Systems (IWIS) | 979-8-3503-0504-3/23/$31.00 ©2023 IEEE | DOI: 10.1109/IWIS58789.2023.10284651

***Abstract*—When creating a deep learning model for estimating the depth of images, constructing a training dataset using stereo images presents a significant challenge. Therefore, using monocular images for depth estimation provides numerous benefits in terms of dataset acquisition. Monodepth2 is one of the prominent techniques for monocular depth estimation. By employing a self-supervised approach, Monodepth2 eliminates the need for ground truth, making the acquisition of the training dataset much easier. Nonetheless, a challenge faced by Monodepth2 is the issue of blurred boundaries in the output depth maps. To address this concern, the paper proposes a modified architecture of Monodepth2, resulting in enhanced accuracy and sharper boundaries in the output depth maps.**

***Index Terms*—Deep Learning, Depth Estimation, Edge Detector**

I. Introduction

Depth estimation is one of the main tasks in computer vision. The goal of this task is to estimate the depth per pixel in images. In recent years, there are many studies about estimating depth. Monodepth2 [1] is represented as one of them. Monodepth2 employs Depth network and Pose network to estimate depth by analyzing the changes in sequential monocular images. This method has the advantage of making it easy to configure the training data set because it uses the Self-supervise method that does not require ground truth. However, in this method, the depth estimation using a monocular image leads to reduced accuracy and blurred object boundaries, as shown Fig.1(b). This problem can make it difficult to distinguish objects. This paper conducted research to solve this problem. The contribution to solving this problem is

- Edge map multiplying during the generation of the depth map to apply weights to the edges.
- Edge maps used for inter-feature comparison in computing Auto-masking loss.

In the process of creating depth, the boundary can be improved by incorporating an edge representation that denotes the object boundaries. This results in a corrected image, as shown in Fig.1(c).

II. Related work

These days, the primary methods for depth estimation using deep learning consist of utilizing either stereo images or monocular images. This paper investigates the use of monocular images, which can be applied in various environments. The method of using a Monocular image is divided into a method of generating a virtual stereo image or estimating depth by predicting and learning changes in the current image based on the previous frame.

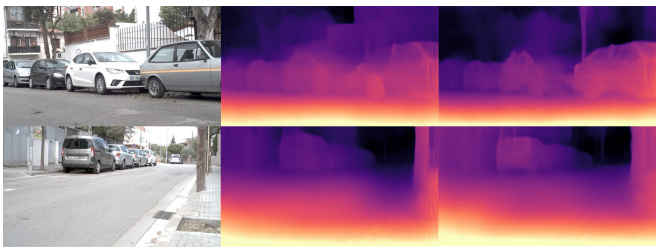

Fig. 1: The input images were sourced from the BIPEDv2 [2] dataset. In the original Monodepth2 image (b), there is a slight blurring of the boundaries of car, whereas in the proposed method image (c), the car's boundaries are more distinct and clear.

A. Depth estimation using Stereo image

Estimating depth from stereo images requires obtaining the disparity information. Disparity refers to the difference in the horizontal position of the same object between the left and right images. In Fig. 2. z represents the depth value for the 3D point (x, y, z) in three dimensions. To calculate the depth value, a proportional equation is

$$x_l : \frac{2}{b} + x = f : x \Rightarrow f(\frac{b}{2} + x) = zx_l \tag{1}$$

$$x_l : \frac{2}{b} + x = f : x \Rightarrow f(\frac{b}{2} + x) = zx_l \tag{2}$$

$$(1), (2) \Rightarrow z = \frac{fb}{x_l - x_r} = \frac{fb}{d(disparity)} \tag{3}$$

Since b(baseline) and f(focal length) are fixed, the disparity becomes crucial for depth estimation.

B. Depth estimation using monocular image

Two popular approaches for estimating depth from monocular images are the virtual stereo image creation method used in Monodepth1 [3] and the use of consecutive images employed in Monodepth2 [1]. In this paper, the focus of the study was on monodepth2, which is known for its superior performance in depth estimation compared to other methods. To estimate

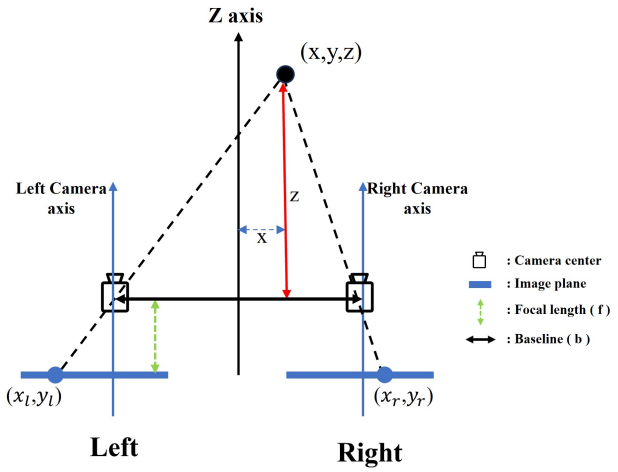

Fig. 2: This illustration depicts the depth estimation of stereo vision. The point (x, y, z) in 3D space is projected onto the left and right image planes through the camera lenses. The horizontal and vertical distances from each camera axis to the projected points are represented by (x_l, y_l) for the left image plane and (x_r, y_r) for the right image plane, respectively. Focal length is distance between image plane and camera lens. Base line is stereo camera interval.

depth from consecutive images, Monodepth2 measures pose changes and uses the measured values to train the model for depth estimation. It is more challenging compared to stereo image-based depth estimation. But it offers the advantage of acquiring a large training dataset more easily.

C. Self-supervised Monocular Training

The main learning methods of deep learning include supervision learning and self-supervision learning. Supervised learning is highly accurate because learning proceeds with ground truth. But data sets for depth estimation are difficult to construct an accurate ground truth, using self-supervised learning that does not require ground truth is very advantageous to secure data sets.

III. Proposed Method

A. Edge Generator

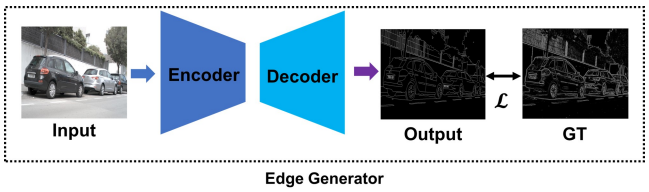

Fig. 3: Structure of Edge Generator. The Encoder compresses the input image using convolution and max-pooling layers and the Decoder restores the compressed image using two transposed convolution layers with ReLU activation. Output channel is set to 1.

The architecture of Edge Generator(EG) is depicted in Fig. 3. The model is structured with an encoder-decoder design. The encoder compresses the input RGB images using 3x3 convolutions and 2x2 max-poolings. Conversely, the decoder decompresses the images using 2x2 transposed convolutions and ReLU activation. During this process, the model output channel was set to 1 to match the channel of the ground truth. For training, the Binary Cross Entropy(BCE) loss function was used. The reason for using edge images is that edges represent the boundaries of objects in an image. Therefore, by incorporating edges into the decoder part of the Depth network, it becomes possible to assign weights to the boundary regions during the upsampling process, enabling the correction and refinement of the boundaries.boundaries in the generated depth map. Furthermore, when using edges for calculating the reprojection error, it enables feature point matching, which helps to find errors more effectively between corresponding points.

B. Modified Architecture of Monodepth2

Monodepth2 network consists of a Pose Network shown in Fig. 4.(b). that calculates image variations, and an encoder-decoder architecture illustrated in Fig. 4.(c). that generates the depth map. The Pose Network employs a pre-trained ResNet-18 [4] to output transformation matrices for input images, while the depth network utilizes ResNet-18 as the encoder and a depth decoder to produce depth maps for the input images. Generated output of I_t depth map is transformed into 3D points using the K^{-1}(Inverse Intrinsic matrix), and the transformation matrices from the Pose Network are applied to convert the 3D points into 3D points of $I_{t'}$. Afterward, the 3D points are transformed back into 2D using the K(Intrinsic matrix), and then the Grid sampling technique is applied to create the $I_{t'\to t}$ image. The learning process aims to minimize the L_p error from Eq. (4).

$$L_p = \sum_{t'} pe(I_t, I_{t'\to t}) \tag{4}$$

In Eq. (4), L_p represents the photometric reprojection error, while p_e denotes the photometric reconstruction error. The p_e is computed using the L1 loss, which involves comparing I_t and $I_{t'\to t}$ and summing up the errors to obtain L_p. Since the boundary of the depth map output as a learning result is blurry, this paper proposes the Fig. 4.(a) model. To enhance the boundaries of the depth map, the Proposed Depth Network(Fig. 4.(c)) was employed in this paper. The Proposed Depth Network compresses the Edge map generated by the EG using 3x3 convolution layer and 2x2 max-pooling layers. Then, it performs element-wise multiplication with the feature map from the decoder of the Depth Network. This approach enhances the clarity of the depth map boundaries by applying weights to the edges.And to improve performance, edge maps were applied to the images used in the Auto-masking loss function. This incorporation allowed for more effective detection of changes between corresponding pixels through feature matching. The Auto-masking loss(Eq. 5.)

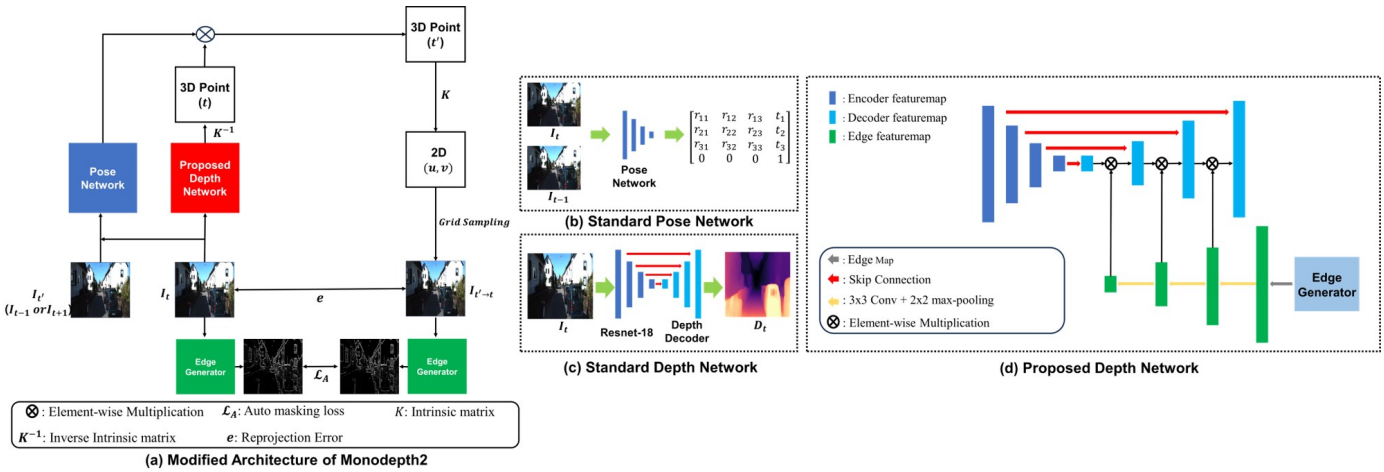

Fig. 4: Structure of Edge Generator. Figure (a) represents the overall approach of the Proposed method, while (b) depicts the structure of the Pose Network, (c) illustrates the Depth Network utilized in Monodepth2, and (d) displays the Proposed Depth Network.

implies that only pixels with a smaller minimum value when comparing the $I_{t'\to t}$ and I_t are taken into account for learning, compared to the minimum value when comparing the I_t and $I_{t'}$. This means that the learning process only considers the parts of the consecutive input images that exhibit changes, i.e., the regions corresponding to object movements.

$$L_A = \min_{t'} pe(I_t, I_{t'\to t}) < \min_{t'} pe(I_t, I_{t'}) \tag{5}$$

prevents the training of datasets that lack changes between images, which could otherwise reduce the accuracy of disparity estimation.

IV. Experiment

A. Dataset

1) KITTI2015: The Modified monodepth2 model was trained using the KITTI2015 [5] dataset, the same dataset utilized by the original monodepth2. The training process utilized 39,810 data samples of image size resolution 640×192 for training and 4,424 samples for validation. The KITTI dataset contains diverse sensor information related to autonomous driving and is widely used for object recognition, tracking, and outdoor depth estimation tasks.

2) BIPEDv2: The Edge Generator employed the BIPEDv2 [2] dataset, comprising 200 training images and 50 test images, all of size 1024×1024. The BIPED dataset contains road images from Barcelona paired with corresponding edge images, both at a resolution of 1280×720. This dataset is widely utilized for edge-related research and serves as a valuable resource for various edge detection tasks.

B. Experiment Detail

1) Edge Generator: The detailed training environment is as follows.

- CPU: Intel(R) Xeon(R) Gold 5218R CPU @ 2.10GHz
- Graphic Card: NVIDEA-A100 40GB * 4EA
- Batch size: 8
- Epoch: 3000
- Loss function: BCELoss
- Optimizer: Adam [6]
- Learning rate: initial learning rate 0.001 with StepLR scheduler

The output results of the test dataset are shown in Fig. 5, indicating that the edge map is well generated.

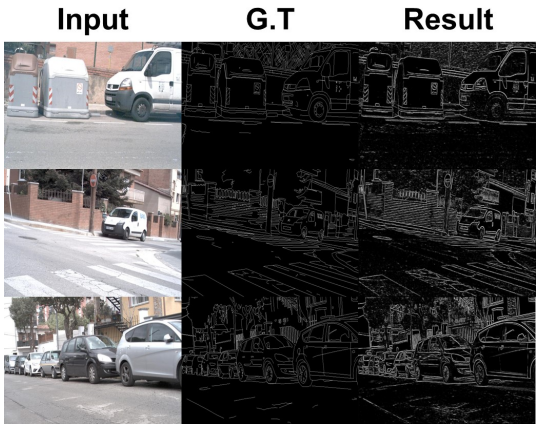

Fig. 5: This is the test result image of the Edge Generator. When comparing the result images with the Ground Truth, it is evident that the Edge Generator accurately produces the edges.

2) Modified monodepth2 model: The detailed training environment is as follows.

- CPU: Intel(R) Xeon(R) Gold 5218R CPU @ 2.10GHz
- Graphic Card: NVIDEA-A100 40GB * 4EA
- Batch size: 12
- Epoch: 14
- Loss function: novel appearance matching loss, auto-masking loss, multi-scale appearance matching loss

	Error			Accuracy		
Method	Abs Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Monodepth2	0.121	4.888	0.198	0.862	0.956	0.981
Proposed method	0.122	4.946	0.194	0.863	0.957	0.982

TABLE 1: The results of comparing Monodepth2 and the Proposed method.

	Error			Accuracy		
Method	Abs Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Monodepth2	0.121	4.888	0.198	0.862	0.956	0.981
Proposed method (w/o Edge map to L_A)	0.121	4.993	0.197	0.867	0.957	0.981
Proposed method (+Edge map to every loss)	0.122	4.96	0.202	0.858	0.953	0.98
Proposed method	0.122	4.946	0.194	0.863	0.957	0.982

TABLE 2: The numerical results of the ablation study

- Optimizer: Adam [6]
- Learning rate: initial learning rate 0.0001 with StepLR scheduler

The experimental results [7] are presented in Table 1. A comparison between the Proposed method and the original monodepth2 revealed an increase in RMSE log and Accuracy. Furthermore, as depicted in Fig. 6, it is evident that the generated depth maps' boundaries became more distinct.

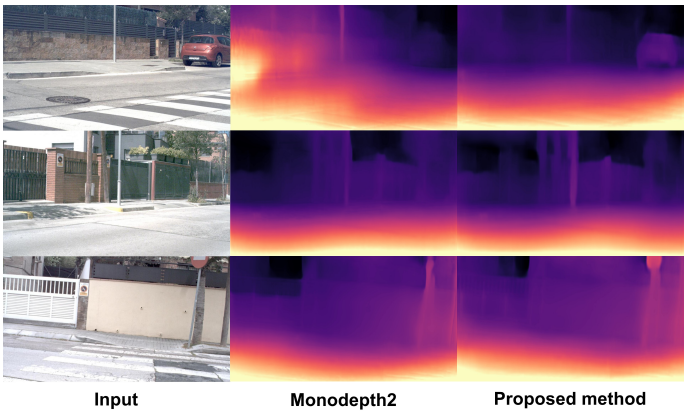

Fig. 6: This is the output depth images of Monodepth2 and the Proposed method. When comparing the Proposed method's images with Monodepth2, it is evident that the boundaries of objects are more distinct.

C. Ablation Study

An ablation study was conducted to assess the optimal results. Table. 2 represents the numerical results of the ablation study. In the Proposed method, it can be observed that when the Edge map is not applied to L_A, the Error increases. On the other hand, using the Edge map in all loss functions reduces the Error, but it leads to lower accuracy. Despite certain increases in Error, the Proposed method exhibits improved accuracy, as evident in Fig. 7, where the depth map boundaries are notably clearer. This indicates that the performance of Monodepth2 has been improved.

V. Conclusion

In this paper, the issue of blurred boundaries in the depth maps generated by the original Monodepth2 model is identified. To address this problem, a Modified Architecture of Monodepth2 is proposed. The Proposed Depth Network within the Modified Architecture assigns weights to the edges of the generated depth maps, resulting in sharper boundaries in the output depth maps. And the use of edge images in L_A enhances the accuracy 0.1% of the results.

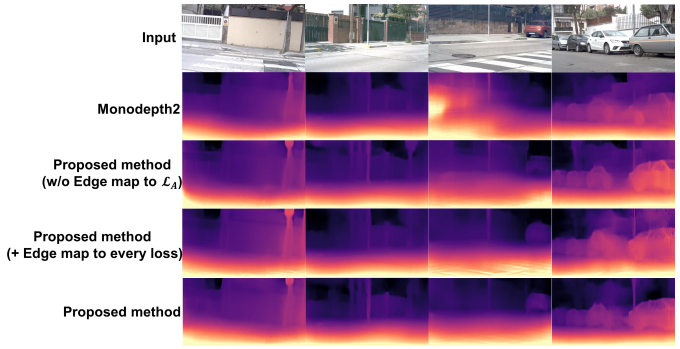

Fig. 7: This image represents the results of the Ablation Study. It can be observed that the depth map with the sharpest boundaries is generated in the Proposed method image.

Acknowledgment

This result was supported by "Regional Innovation Strategy (RIS)" through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(MOE)(2021RIS-003)

References

[1] Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J Brostow. Digging into self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3828–3838, 2019.

[2] Xavier Soria, Edgar Riba, and Angel Sappa. Dense extreme inception network: Towards a robust cnn model for edge detection. In *The IEEE Winter Conference on Applications of Computer Vision (WACV '20)*, 2020.

[3] Clément Godard, Oisin Mac Aodha, and Gabriel J Brostow. Unsupervised monocular depth estimation with left-right consistency. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 270–279, 2017.

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[5] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[6] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[7] Cesar Cadena, Yasir Latif, and Ian D. Reid. Measuring the performance of single image depth estimation methods. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4150–4157, 2016.