Proposal for Monocular Depth Estimation in Single Image

Shuai Wang(swang86) Wenhao He(wenhaohe)
Bochun Deng(bochunde) Ninghui Jin(ninghuij)

April 17, 2024

1 Abstract

Suppose a photo contains a series of objects. These objects change from 3D to 2D in the process of taking pictures. Much of the information about these objects is lost when converted to 2D images. For example, two originally parallel lines will become two oblique lines intersecting at the vanishing point during the process of converting from 3D to 2D. In computer vision, these missing parameters play a crucial role in restoring 3D images. One of the most important parameters is the distance between the photographer and the object. Due to the influence of parameters such as depth of field and focal length, the important parameter of the distance from the object when taking pictures has been lost in the process of being two-dimensional. This project mainly uses deep learning algorithms to complete the distance estimation of objects in the image.

2 Introduction

This project estimates the distance of objects in a 2D image from the photographer. The dataset[4] is a sequence of images with specific objects, it works for both indoor and outdoor scenes without any domain specificity. Each photo comes with different parameters such as depth of field and focal length. Every photo is a collection of pixels. Each pixel has three numbers for red, green, and blue. Together they will represent the color of a pixel. Put these pixels in a two-dimensional matrix to finally form the entire image.

In deep learning, a model that works well for image processing is a convolutional neural network. It simulates human vision by creating multiple layers of convolutional and pooling layers. It can use algorithms to complete feature extraction and abstraction to complete a higher-level understanding of images. A complete convolutional neural network consists of convolutional layers, pooling layers, and fully connected layers. Convolutional layers are mainly used for feature extraction. Pooling layers are mainly used to reduce the photo size and preserve important information. The pooling layer can improve the processing

efficiency of the computer on the image data. The fully connected layer is mainly used for the final distance estimation. Since distance estimation is a prediction of continuous data, the fully connected layer also plays a vital role in it.

3 Methodology

Figure 1: The neural network architecture with the outputs of the layers

This project will use image data to train a convolutional neural network. The convolutional neural network adopts an encoder-decoder architecture[1]. The Encoder part is a pre-trained model Efficientnet-b5[2]. The Decoder part has a total of 12 convolutional layers. There are 2352 nodes in the first layer. After that, the number of nodes in the layers decreases until the 12th layer has 128 nodes. The activation function of the model is Leaker ReLu. The parameter learning rate in the model is 0.0001, and the batch size is 1. The training set is a series of pictures with objects. The label is the shooting distance of the photographer. The image data is divided into training and testing sets. The training set is used for model training and parameter tuning. The test set is used to test the prediction results of the trained model.

The other group is the comparison group. The comparison group was completed by Nguyn Th Thanh Hoà of Kaggle. The comparison group contains 7 convolutional layers and 3 pooling layers. The nodes of the convolutional layer are incremented from the initial 50 until the seventh layer contains 120 nodes. The activation function of the comparison group is ReLu. The size of the pooling layer is 2 by 2.

4 Analysis

The figure 2 shows the distance test of the trained model on several pictures. When the pixel is yellow, the distance is larger. Pixels are closer to the photographer when they are red and black.

The figure 3 shows the impact of the parameter batch size on the loss of test results. The y-axis is the loss of the test result. The x-axis is different batch sizes. The batch size indicates the size of the amount of data to be trained in a

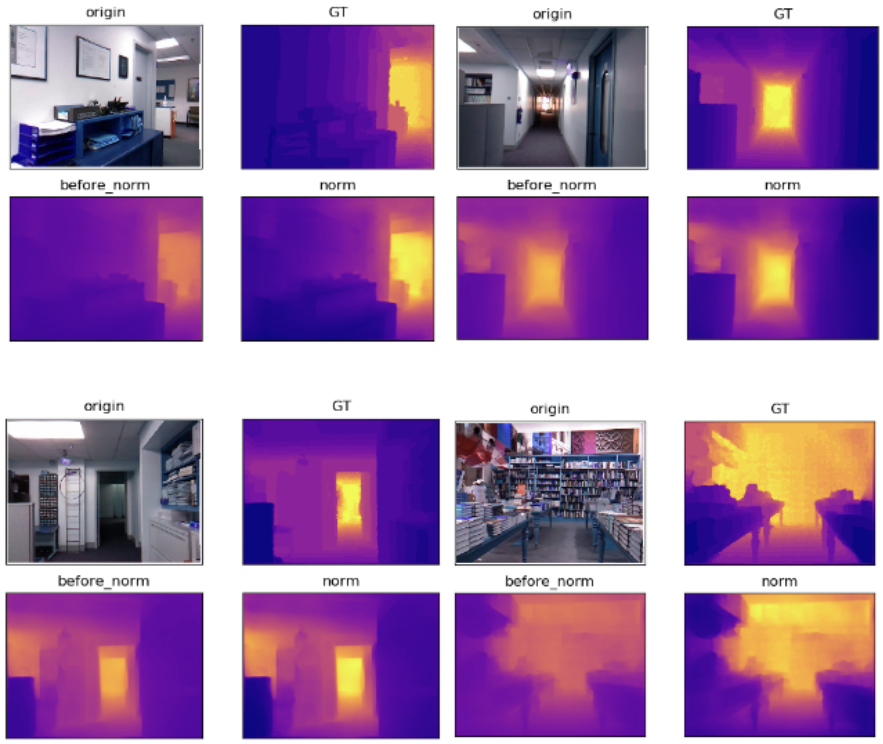

Figure 2: Test Result

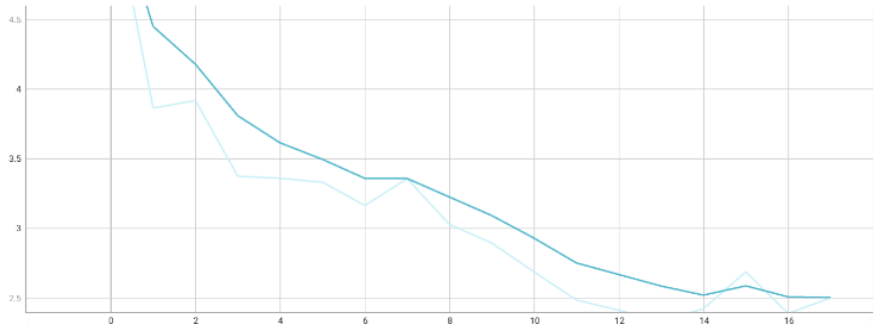

Figure 3: Loss/Batch; 4000 pictures per batch

forward/backward pass. The batch size plays a vital role in model optimization. As the batch size increases, the loss decreases from the initial 4.5 to 2.5. It can

be seen from the figure that for the model in the project, the optimal batch size is 16.

5 Discussion

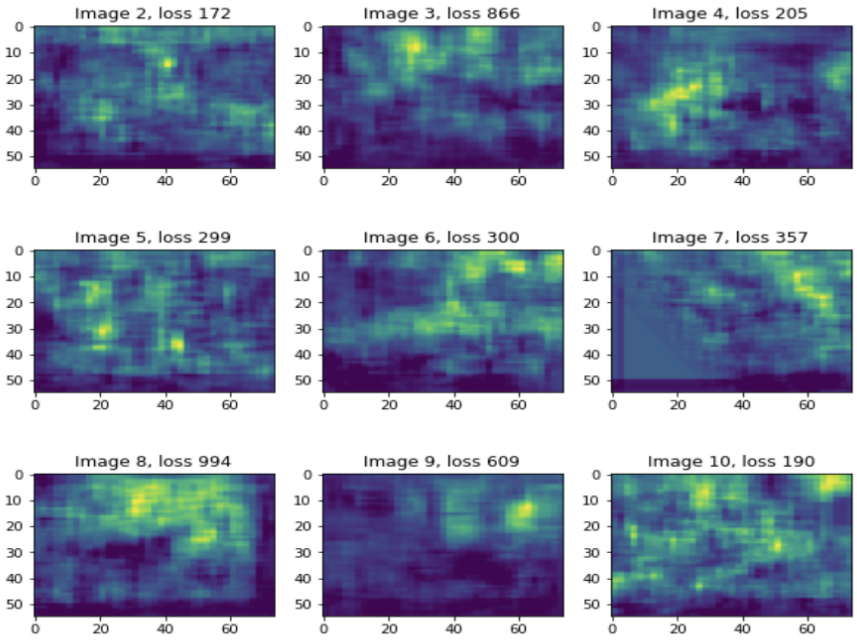

Figure 4: Result of image depth estimation

The picture above is the result of image depth estimation from Kaggle by Nguyn Th Thanh Hoà[3]. This is the result of a comparison group. As can be seen from the figure, its distance depends on the final RGB value obtained. In the case of a larger distance, the pixel will be more yellow. On the contrary, it is closer to black. Image 8 has the largest loss. The reason is that there is an object very close to the photographer on the straight line that the photographer focuses on. Also behind this object is the furthest point in the picture. Under the action of the convolutional neural network pooling layer, the information where the two intersect is seriously lost. This is the reason for its large loss. In distance estimation, errors are prone to occur when distant objects and close objects are on the same line.

Compared with the comparison group, this project does not use a lot of pooling layers to process images. The loss of information is not as much as

that of the comparison group. At the same time, the result obtained by the project is that the processing efficiency is not as good as that of the comparison group but with smaller losses. The batch size used in the comparison group is 32. Compared with the model in the project, the comparison group can use a larger batch size to reduce the loss. This approach also comes at a price. The comparison group needs more computation power to complete the training of 32 batch size. This reduces training efficiency. The models in the project focus more on building complex convolutional neural networks to improve accuracy. The model of the comparison group uses a larger batch size to improve the accuracy.

6 Conclusion

This project mainly uses convolutional neural networks to estimate the distance between the object and the photographer in the picture. When using convolutional neural networks, the batch size of training plays a big role. The error between the test result and the real value obtained by the smaller batch size is larger. When the batch size reaches 16, the error reaches the optimal value. For the comparison group, the model uses a relatively simple neural network structure and a large batch size. The model in the project uses a more complex model and a smaller batch size. Both methods can complete the estimation of the distance between the photographer and the object in the image while maximizing efficiency.

7 References

[1] Liu, J., Zhang, Y. High quality monocular depth estimation with parallel decoder. Sci Rep 12, 16616 (2022). https://doi.org/10.1038/s41598-022-20909-x
[2] Mingxing Tan and Quoc V. Le. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. ICML 2019. https://arxiv.org/abs/1905.11946
[3] Silberman, N., Kohli, P., Hoiem, D., Fergus, R. (2012). NYU Depth Dataset V2. Retrieved May 5, 2023. https://cs.nyu.edu/~silberman/datasets/nyu_depth_v2.html
[4] Nguyn, T. T. H., Ho, Q. H., Vi, N., Nguyen, T. V. A., Nghia, L. T. (n.d.).Depth-Estimation-from-Single-Image$_CNN.Kaggle.RetrievedMay5, 2023.Depth-Estimation-from-Single-Image_CNN$