

# **Depth Estimation: Dnet Model**

## **1.Introduction:**

Calculating the object distance from a camera is a fundamental machine vision problem. To calculate distance of an object from camera by estimating the size of the object from an image, given only the raw image file and no metadata except for the resolution and number of pixels and then interfacing it with Mask RCNN to get the distance of the target object.

## **2.Methods and reference:**

Paper followed: “Toward Hierarchical Self-Supervised Monocular Absolute Depth Estimation for Autonomous Driving Applications”.

This paper has an idea of implementing self-supervised method with DNet architecture. At first relative depth estimation is done with the dense connected prediction (DCP) that hierarchically combines features in different levels and handles local gradients, after which scale recovery is done to get ground-level depth. We estimate scale factor for current relative depth and from there absolute depth is calculated pixel-wise.

- Github repo: <https://github.com/TJ-IPLab/DNet>

## **3.Basic Idea:**

DNet architecture is followed here for testing models. DNet is a self-supervised monocular depth estimation pipeline that exploits densely connected hierarchical features to obtain more precise object-level depth inference and uses dense geometrical constraints to perform scale recovery. We evaluate the depth of custom images by testing on a pretrained model. Here we use custom images to observe how the architecture is performing. We evaluate the relative depth from that model and perform scale recovery using dense geometrical constrain module. Then we estimate the absolute depth. Then we use Mask RCNN to detect the particular object in the image and calculate mean absolute depth value, thus considering it as the depth from camera.

## **4.Current Status:**

We tested on several indoor and outdoor images. Camera height is a crucial parameter in this model. We tested on different images with variant distances and different camera heights to check how the deviation in distance is changing with increase or decrease of camera height. In the case of indoor images with very less distance, the model is giving moderate results. We tested further on outdoor images with different objects, here those images where Mask RCNN is identifying more than one mask within the object is performing really badly. We found that for

an instance with the exact camera height our results were giving slightly high deviations however as we decreased the height, the predicted distance and the deviation was decreased by a range. As mentioned in the paper, they have used a statistical method for estimating camera height which is unknown and not clearly portrayed. Scale factor here is determined through comparison between the given and estimated camera height. As there is an issue in getting the camera height and the proper measures are unknown, we are getting anomalies and errors in distance prediction. Further, low and high resolution of camera pictures is an issue here. We have used mobile cameras while they have used on-board cameras to capture the images. Different factors are there due to which the deviation is varying. A proper conclusion could not be made from these verdicts.

## **5.Observations:**

From the above observations it is clear the model is giving good and bad results. The Only variable that can affect this is the camera height. Both indoor and outdoor images were used. In both the cases the with respect to various camera height the model is performing differently. The evidence for this is in the case of Motorcycle, even though the camera height is between 150-165 cm for it, the model is performing very badly. But when we gave the metric as a random value in this case at 45, the deviation decreased from around 300 % to less than 10 % even though 45 wasn't the correct height.

## **6.Output:**

<b>Object Tag</b>	<b>Camera Height(appox)</b>	<b>Actual Distance</b>	<b>Predicted Distance</b>	<b>Deviation</b>
Stool	13 cm	125 cm	131 cm	4.8 %
Bottle	13 cm	90 cm	109 cm	21.1 %
Hydrant	38 cm	300 cm	338 cm	12.6%
Hydrant	37 cm	500 cm	260 cm	48%
Motorcycle	165 cm	500 cm	19.11 m	294 %
Motorcycle	Random (45 cm)	500 cm	5.37 m	7.4 %
Motorcycle	165 cm	6 m	20.45 m	240 %

Motorcycle	Random (45 cm)	6 m	5.58 m	7 %
------------	----------------	-----	--------	-----



**ACTUAL DISTANCE = 0.90m**

**PREDICTED DISTANCE = 1.09m**

**DEVIATION = 21.1%**



**ACTUAL DISTANCE = 1.25m**

**PREDICTED DISTANCE = 1.31m**

**DEVIATION=21.9 %**



**ACTUAL DISTANCE = 3.00 m**

**PREDICTED DISTANCE = 3.38 m**

**DEVIATION=12.6 %**



**ACTUAL DISTANCE = 5.00 m**

**PREDICTED DISTANCE = 2.6 m**

**DEVIATION=48 %**





**ACTUAL DISTANCE = 5.00 m**

**PREDICTED DISTANCE = 19.11 m**

**DEVIATION=294 %**



**ACTUAL DISTANCE = 6 m**

**PREDICTED DISTANCE = 5.58 m**

**DEVIATION=7 %**

### **8.Conclusion:**

The Camera height we are giving may not be the exact height. For the same object at different height it is giving varying values. I am not able to figure out a way for this. More research on how they are analysing this camera height is needed in order to move forward.