CPE 393 Coding in AI

Depth Estimation Using CNN architecture

Member

Nattapat	Yuvasuta	59070501028
Niti	Buesamae	59070501047
Pattharapong	Chotechuang	59070501101

1 Abstract

Depth detection is a fascinating task with enormous opportunity to be adopt into various application such as mixed reality application, image blur etc. In this work we attempt to build models to do such that using technologies and ideas proposed by many academic papers. Our work could also be used to help reduce hardware cost on many task by using pure software with help of deep learning technology.

We build 3 different neural network variants, the first architecture we tried is CNN encoder-decoder with fully connected layers in the middle. During training period we found that the model can perform well to overfit training data but fail to predict depth on testing data due to usage of fully connected layer as the main part to learn perceptual features extracted by CNN encoding layer as the architecture fail to generalize it prediction on both set of data.

The second and the third is Multi-scaling CNN. We found better performance in the same dataset on these variant. The architecture separate 2 convolutional works into 2 scale that looking for difference level of perceptual representation. In the first scale the CNN network focus on mapping coarse detail in the picture (eg. the scene area, wall angle etc.) into initial depth that show the whole scene depth estimation. The second level is finer. It maps features from smaller objects in the scene into finer depth distribution that show detailed estimation of objects in different position in the scene.

Training with 2 of these, this variant of CNN perform better than the simpler variant and yield acceptable result in term of metrics used to compare it with state-of-the-art models.

2 Introduction

Our topic is image depth estimation model using encoder-decoder with CNN. The main goal is to estimate depth values of a single image input and generate depth map as an output. We use NYU V2 as our dataset. We hope that our study will produce a result that is at least close to the baseline of the state-of-the-art and make a new path of depth estimation modeling.

We proposed 3 different Convolutional model to predict depth field image from single RGB image. To do that we adopt 2 techniques used in academic papers to extract perceptual features from images and use them as inputs to generate depth prediction just like how human estimate the environment's depth by looking at the shape, line, color, and light in comparison with the real depth after actually touching or feeling it.

First architecture is CNN encoder-decoder merged with fully connected layer in the middle. The idea behind this model is quite simple. As the image data come into the model in form of three channel pixel data, the model use CNN encoding layer to map raw pixel data into rich perceptual representation with collections of feature vector and flatten it into a tensor to be used as inputs for fully connected layer sandwiched inside. As the data has been encoded and processed, it is finally being upsampled into a new image that have only 1 channel showing depth prediction in each pixel of the image.

The second architecture is Multi-scaling CNN. The idea behind this model is to capture the information of overall image and refine it to finer details with two models including coarse and fine model. As images are fed into the model, a pre-trained coarse model evaluate the image by convoluting the raw pixel data and feed them into dense layers to get coarse depth prediction which will then be merged with the finer model prediction later. At the same time, the fine model will feed raw data, convoluting them and merge them with coarse predicted depth as mentioned before convoluting them again to get final result that should represent details of edge and curve of the depth map predicted as well as the whole scene depth values.

3 Theory

3.1 Encoder-Decoder

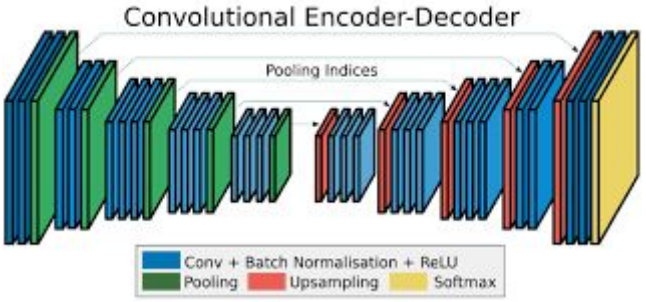
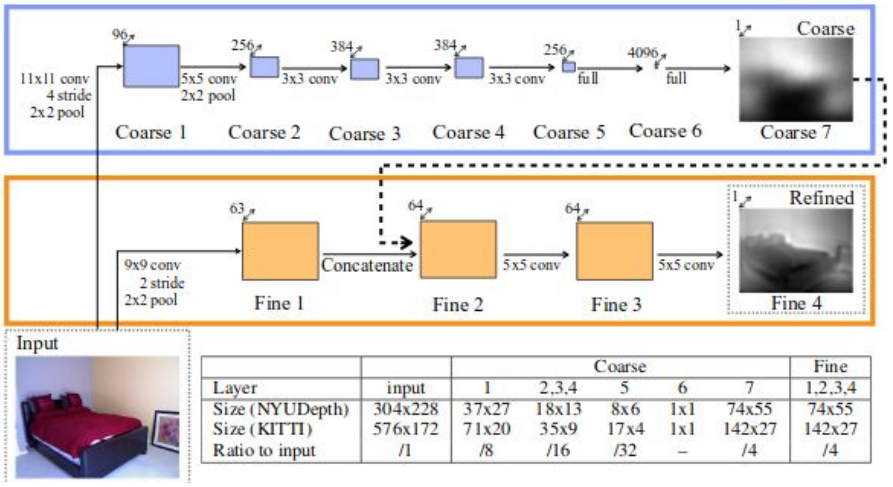


Figure 1. Convolution Encoder-Decoder Architecture
reference: http://www.cs.toronto.edu/~urtasun/courses/CSC2541/07_segmentation.pdf

Encoder can be used to encode feature of raw image (RGB color) to a rich presentation of a collection of a feature vector. Decoder takes the features that produces from encoder to produces an output and map output back to the raw format(image pixel). In principle, the encoder and the decoder scheme is arbitrary, the architecture can be adopt to use with CNN's, RNN's, multilayer perceptrons etc.

3.2 Multi-scale



		Coarse					Fine
Layer	input	1	2,3,4	5	6	7	1,2,3,4
Size (NYUDepth)	304x228	37x27	18x13	8x6	1x1	74x55	74x55
Size (KITTI)	576x172	71x20	35x9	17x4	1x1	142x27	142x27
Ratio to input	/1	/8	/16	/32	–	/4	/4

Figure 2. Multi-scale architecture, Eigen et al.[6]

Multiscale methods, in which a dataset is viewed and analyzed at different scales,are becoming more commonplace in machine learning recently and are proving to be valuable tools. At their core, multiscale methods capture the local geometry of neighborhoods defined by a series of distances between points or sets of nearest neighbors. This is a bit like viewing a part of a slide through a series of microscope resolutions. At high resolutions, very small features are captured in a small space within the sample. At lower resolutions, more of the slide is visible, and a person can investigate bigger features.Main advantages of multiscale methods include improved performance relative to state-of-the-art methods and dramatic reductions in necessary sample size to achieve these results.[9].

4 Experimental Design and Results

4.1 Encoder - Decoder CNN with Fully connected

4.1.1 Model Architecture

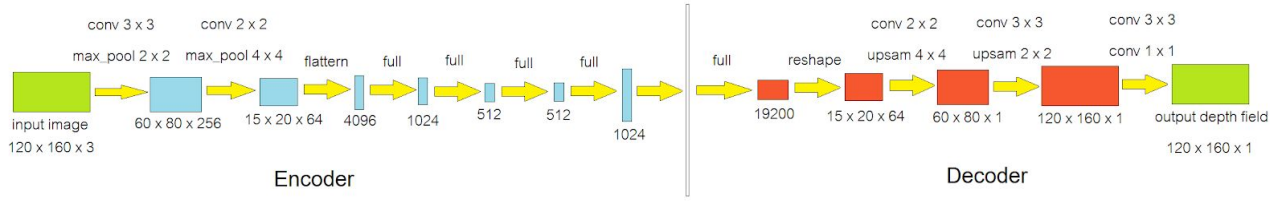


Figure 3. Encoder - Decoder with CNN-Fully connected

Our network consists of two parts: encoder and decoder. Encoder contains contains two feature extraction layers of convolution and max-pooling, followed by four fully connected layers. The input layer receives an input image size of 120 x 160 pixels with 3 color channels and depth field size of 120 x 160 with 1 channel. An output from encoder is passed through decoder given an depth field output of current input image. A funnel like architecture is used to merge unmeaningful feature into meaningful feature later on.

Scaled exponential linear units activation function with glorot uniform weight initializer is used for every fully connected layers in our model. For optimizer, we use Adadelta with a learning rate of 3.0 and mean square error for loss function.

4.1.1.1. Encoder

Started with a 256 features, 3x3 stride 2D convolution layer followed by max pooling layer with 2x2 stride in order to reduce computing time and power along with feature extracting and reduce insignificant feature. Same with another two 64 features, 2x2 stride 2D convolution layer and 2x2 stride max pooling layers. Flatten layer is then used before fully connected layers. Zero padding is applied to every convolution layers including those on decoder. Fully connected layer is used to predict depth by using extracted features from previous convolution layers using number neurons of 4096, 1024, 512, 512, and 1024 respectively. A bias with random uniform initializer is used on the first fully connected layer.

4.1.1.2. Decoder

The output from encoder is then resize to 15 x 20 with 1 channel. Then we use convolution layer with 2 x 2 stride, upsampling with 4 x 4 stride and then follow by convolution layer with 3 x 3 stride and upsampling with 2 x 2 stride to resize the image to prefered original size. Lastly two convolution layers with 3 x 3 stride and 1 x 1 stride is applied.

Our model perform surprisingly good on training data but a pixelated picture is obtained as a result from deconvolution processes.

4.1.2 Training Result

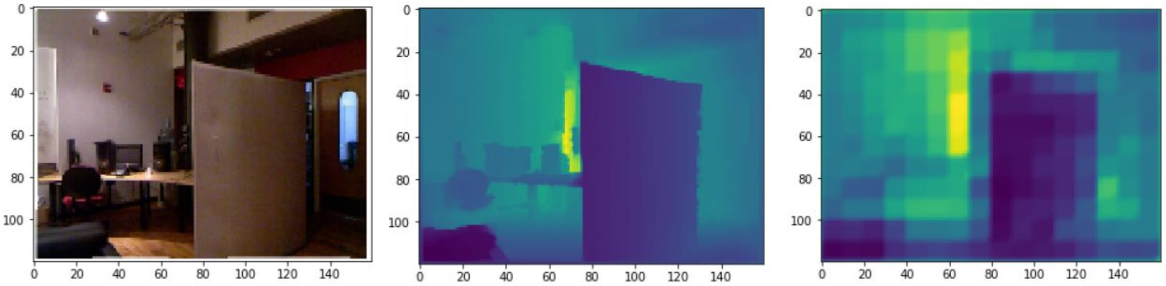


Figure 4. Train image (Left) Depth truth (Middle) Prediction (Right)

4.2 Multi-scale Deep Network

4.2.1 Multi-Scale Deep Network Architecture

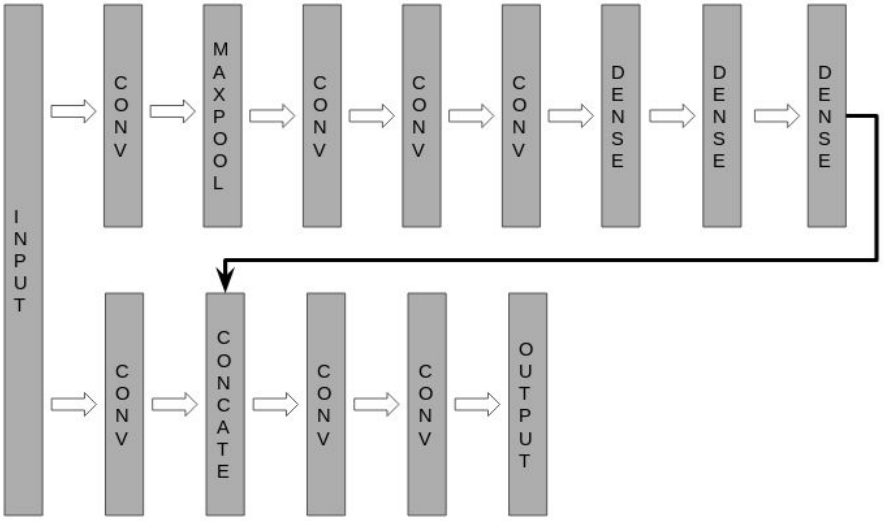


Figure 5. Multi-Scale DNN Architecture

We train 2 different variation of this architecture.

Coarse Model : The purpose of this model is to identify rough depth of the overall image by using convolution, pooling and dense layers. The architecture of this model included 128 features 20 x 20 stride convolution layer, max pooling layer with 2 x 2 stride, 512 features 10 x 10 stride convolution layer, max pooling layer with 2 x 2 stride, two 400 features with 5 x 5 stride convolution layer and 256 features with 5 x 5 stride convolution layer to exact and sizing down an image. Then follow by two dense layer with 400 and 256 units respectively with ReLu activation function and he_uniform weight initializer.

Fine Model : The purpose of this model is to sharpen or refine the edge of shape in the image so dense layers are not necessary for this model as it mainly

aim to capture the hidden detail. The layers that are used in this model included 63 features with 5 x 5 stride convolution layer and two 64 features with 5 x 5 stride convolution layers. Both of activation function and weight initializer that we used are same as coarse model.

4.2.2Result

4.2.2.1 Multiscale #1 (Best fit)

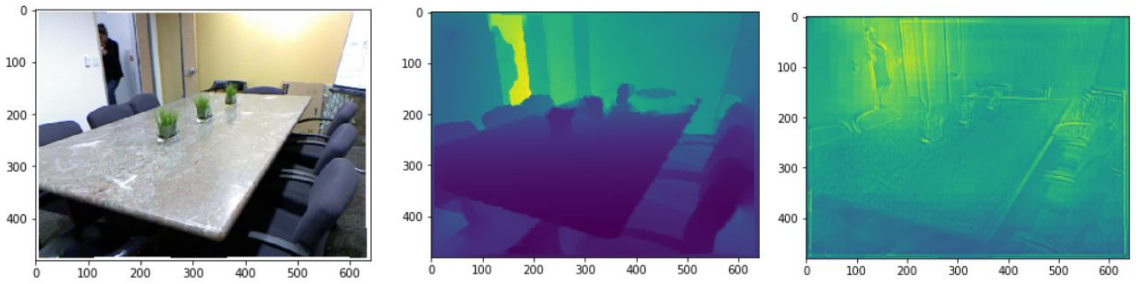


Figure 6. Test image (Left) Depth truth (Middle) Prediction (Right)

This model show poor distribution of depth value for small object but can show acceptable overall depth prediction of the whole scene.

4.2.2.2 Multiscale #2 (Overfit)

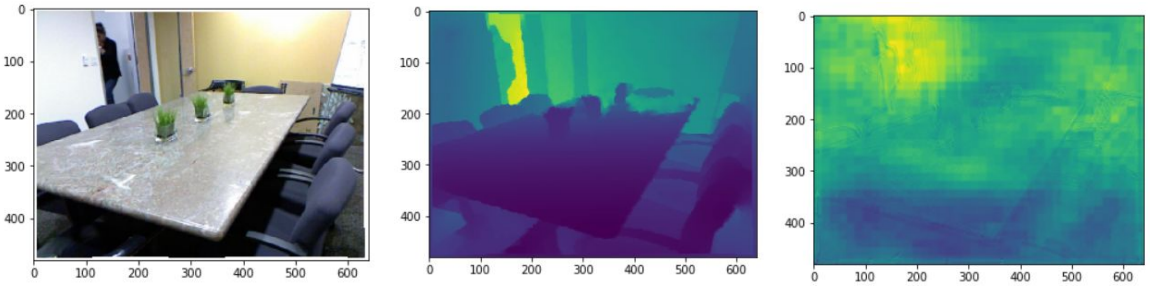


Figure 7. Test image (Left) Depth truth (Middle) Prediction (Right)

This model show better distribution of depth value on small object and easier to identify finer detail of object in the scene however due to overfitting this model lose its resolution so the depth predictions are more pixelated.

4.3 Data augmentation

We augment the training data from NYUDepth V2

- *Flips* : Input and target are horizontally flipped and vertically flipped
- *Noise* : Input are noise

4.4 Loss function

Mean square error as a training loss. Inspired by Eqn. 1

MSE: $\frac{1}{|T|} \sum_{y \in T} ||y - y^*||^2$ (1)

4.5 Comparisons

We measure these following metrics used in our baseline references.

Abs Relative difference : $\frac{1}{|T|}\sum_{y\in T}||y-y^*||^2/y^*$ (2)

RMSE : $\sqrt{\frac{1}{|T|}\sum_{y\in T}||y-y^*||^2}$ (3)

	abs relative difference	RMSE
Eigen et al.[2]	0.215	0.871
Liu et al.[5]	0.230	0.824
Cao et al.[1]	0.232	0.819
Our Model(CNN+FC)	**0.371**	**1.04**
Our Model(Multi-Scale #1)	**0.325**	**0.937**
Our Model(Multi-Scale #2)	**0.329**	**0.812**

Table 1: Comparision on the NYUDepth dataset

5 Conclusion

In this work, we tried fitting several CNN model base on models used in various academic paper to predict depth of single image data. We used many succeeded architecture as a guideline to design our own version to predict on the NYU V2 dataset. The result show that the multi scale convolutional architecture is the optimal choice for predicting the depth base on our knowledge constraint and dataset and converge faster than plain CNN + Fully connected architecture that has limited capabilities to generate complex output image. Though our multi-scale convolutional network still perform poorer than the state-of-the-art in term of real prediction result but proved to be feasible to perform better giving more tuning and modification. At the same time, this work give us an opportunity to explore techniques and theories in related field of neural network and image processing to solve real world challenges which benefit long term study and development and can be used as a baseline for further work in depth prediction.

References

[1] Y. Cao, Z. Wu, and C. Shen. Estimating depth from monocular images as classification using deep fully convolutional residual networks. IEEE Transactions on Circuits and Systems for Video Technology, 2017.
[2] D. Eigen,C. Prhrsch and R. Fergus. Depth Map Prediction from a Single Image using a Multi-Scale Deep Network.arXiv preprint arXiv:1406.2283.2014
[3] J. Hu,M. Ozay, Y. Zhang and T. Okutani. Revisiting Single Image Depth Estimation: Toward Higher Resolution Maps with Accurate Object Boundaries. arXiv preprint arXiv:1803.08673,2018
[4] A. Liaw and M. Wiener. Classification and regression by randomforest.R news, 2(3):18–22, 2002
[5] F. Liu, C. Shen, and G. Lin. Dd G. Lin. Deep convolutional neural fields for depth estimation from a single image. In CVPR, pages 5162–5170, 2015
[6] X. Ma,Z. Geng and Z. Bie. Depth Estimation from Single Image Using CNN-Residual Network. cs231 Stanford.edu,203,2017
[7] T. Koch,L. Liebel,F, Fraundorfer and M, Körner. Evaluation of CNN-Based Single-Image Depth Estimation Methods. arXiv preprint arXiv:1805.01328,2018
[8] Y. Wu, S. YIng and L. Zheng. Size to Depth: A New Perspective for Single Image Estimation. arXiv preprint arXiv:1801.04461,2018
[9] Multiscale_modelting. retrive from https://www.kdnuggets.com/2018/03/multiscale-methods-machine-learning.html