Predicting depth image from a single RGB image

Based on the paper ***Learning Fine-Scaled Depth Maps from Single RGB Images***
By - Jun Li, Reinhard Klein & Angela Yao

Neha Das neha.das@tum.de
Saadhana Venkataraman saadhana.venkataraman@tum.de
San Yu Huang ga59hoc@mytum.de
Sumit Dugar sumit.dugar@tum.de

TECHNISCHE UNIVERSITÄT MÜNCHEN

INTRODUCTION

Purpose and motivation for generating depth maps from RGB images

- Depth images provide richer representations of objects and the environment
- May lead to improvements in prediction tasks due to additional information
- May help in developing additional applications such as 3D modeling

Major Challenges and Related Works

- The task is inherently ambiguous, with a large source of uncertainity emanating from the overall scale
- Predicting fine-scaled features is particularly hard, as observed in some of the previous works in this area[2]

DATASET

For training the model, we use the dataset from RMRC indoor depth challenge[3] which is a subset of the NYU Depths Dataset V2[4]. The total dataset consists of ~4000 RGB-Depth pairs for training. Since, the dataset is too small, we augment the data by applying transformations such as horizontal and vertical flips.

NETWORK ARCHITECTURE DESCRIPTION

Our work in this project is largely inspired by the findings of the paper "Learning Fine-Scaled Depth Maps from Single RGB Images"[1]. As in the paper, we tackle various challenges posed by the previous work and the general formulation of the problem by incorporating the following structures in our network architecture:

Multiple Scales:
This network accumulates information from the RGB image on three different scales which are then reconciled to give us depth images with better resolution.

- *Scale 1:* Scale 1 accumulates global inforrmation from the RGB data through a VGG-16 network with 2 fully connected nets at the end. This layer is primarily responsibe for the underlying structure of the depth map
- *Scale 2:* predicts a coarse map that is nearly 1/4th of the input size. This scale is designed to get more local details than scale 1
- *Scale 3:* predicts a depth map with finer details and higher resolution.

Set Loss for regularization:
To avoid overfitting, a unique form of regularization is imposed. We invert the predicted depth maps (D_i, D_j) for the flipped images by applying the inverse augmentation function (g_{ij}) and minimize the mean squared difference (E) between the various predicted images of a set.

$$L_{\text{set}} = \frac{1}{N-1} \sum_i \sum_{j, j \neq i} E(D_i, g_{ij}(D_j))$$

Skip Layers
Two skip layers are added between the scales 1 and 2. They radically improve the time it takes to converge for the network.

LOSS AND EVALUATION METRICS

MSE Loss:
We minimize a pixelwise mean square error between the predicted and actual depths in order to train our network. In addition to this, we also use a regularization loss – Set loss as outlined in the previous section to prevent our network from overfitting

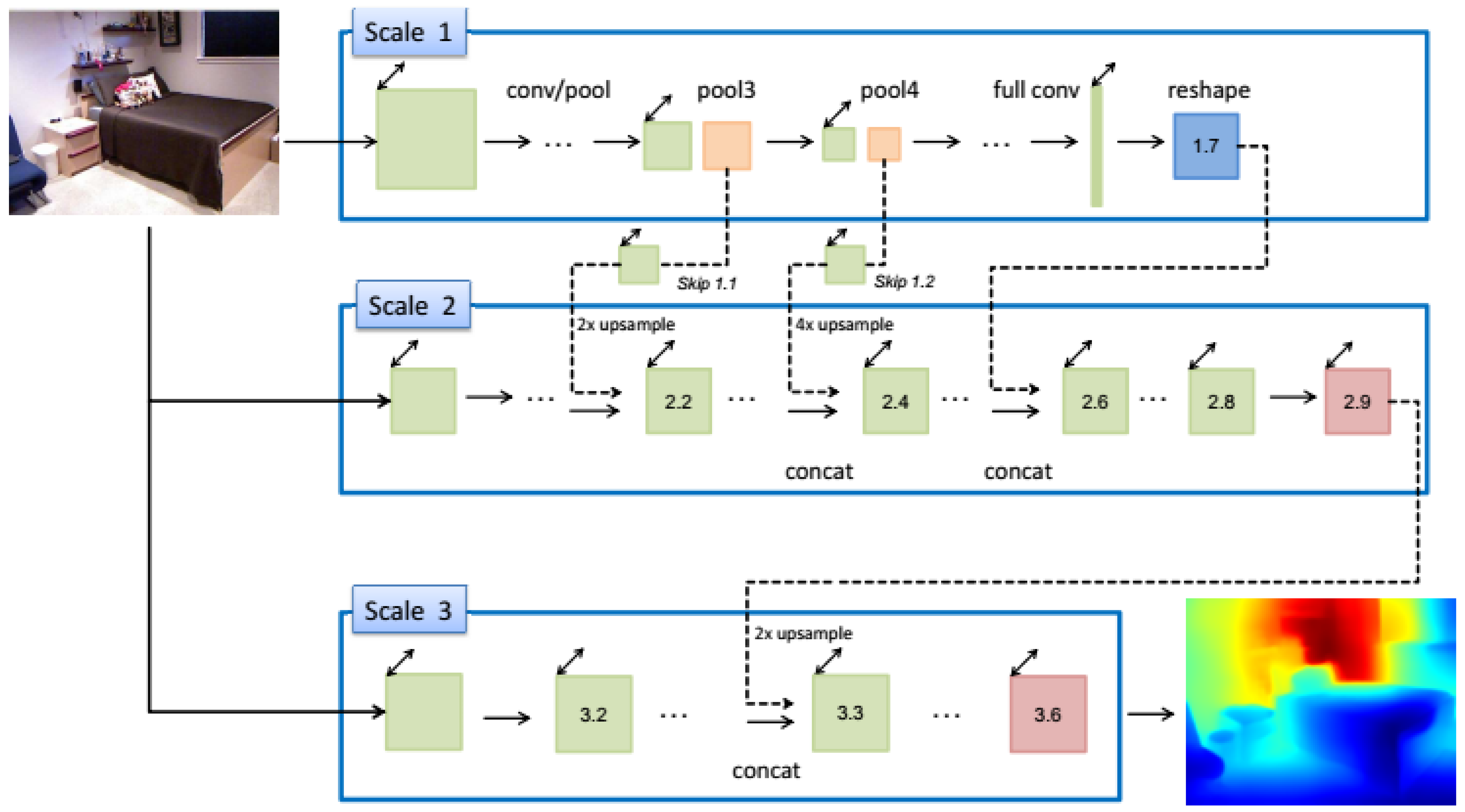
We also attempted to minimize a root mean square loss, but the results were not comparable to MSE Loss

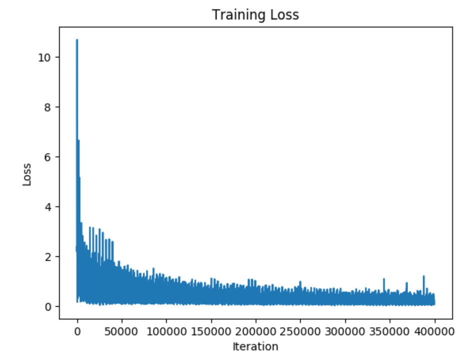
Thresholded Accuracy:
In order to evaluate the performance of the network, we set the accuracy function to the following:

$Max(d_i^{gt}/d_i, d_i/d_i^{gt}) = \delta < thr$

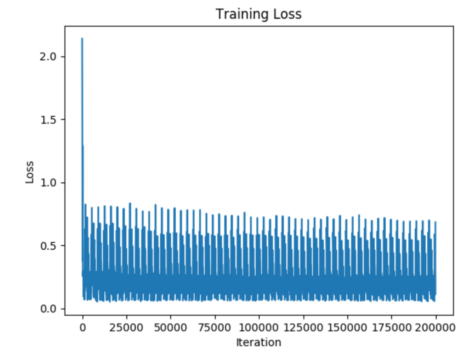
Here *thr* is referred to the range of the allowed relative depth values for the predicted and actual depth images.

ARCHITECTURE AND EXPERIMENTAL RESULTS

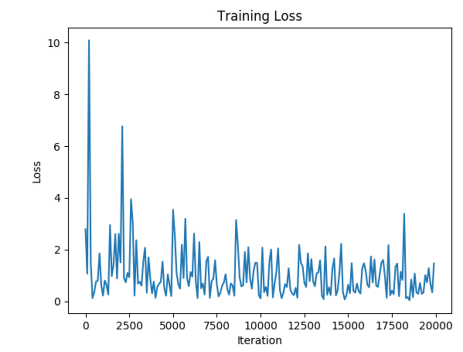




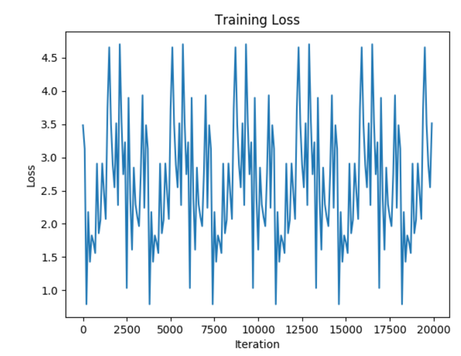
Experiment 1



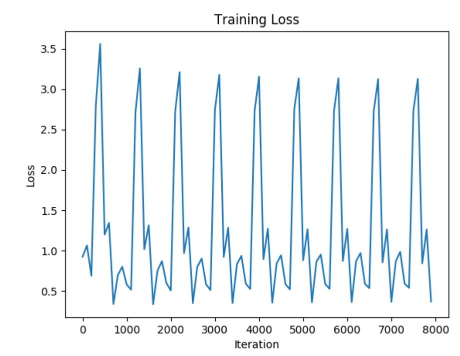
Experiment 2



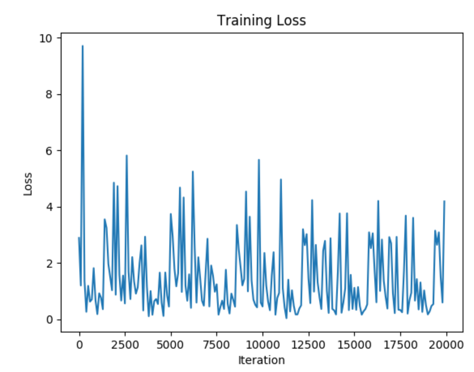
Experiment 3



Experiment 4



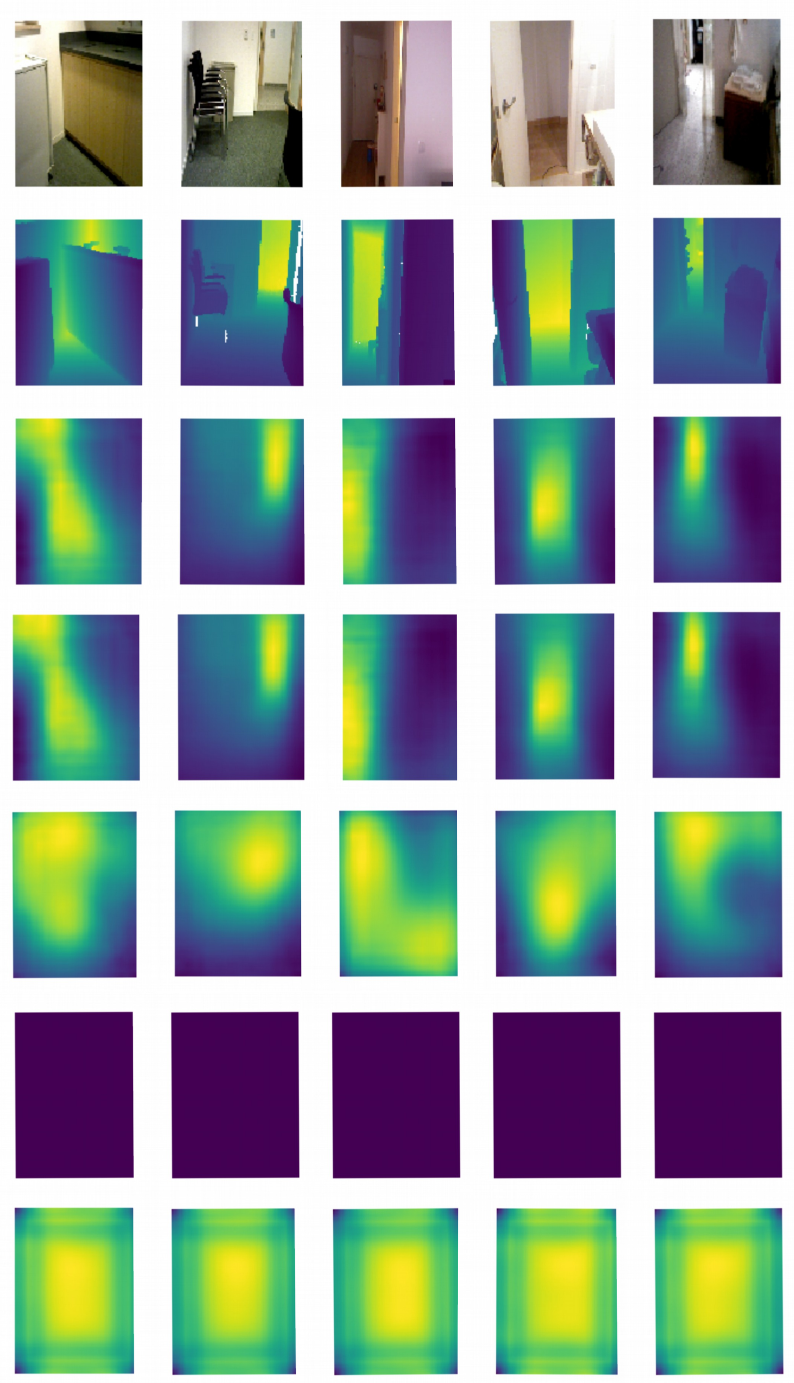
Experiment 5



Experiment 6

EXPERIMENTS

RGB Images
Actual Depth Images
Predicted Depth Images *Experiment 1*
Predicted Depth Images *Experment 2*
Predicted Depth Images *Experiment 3*
Predicted Depth Images *Experiment 4*
Predicted Depth Images *Experiment 5*



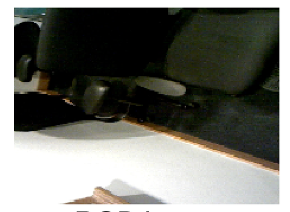
	Experiments	Observations
1)	• Trained scale 1 & 2. • Used mean squared loss and set loss. • Epochs = 100 • Batch size = 1	• Training time = 16 hours. • Training Acc = 0.877 • Testing Acc = 0.514 • Able to learn global depths. • Not able to learn local depths and object structures that well.
2)	• Trained scale 3 while keeping scale 1 &2 fixed. • Used mean squared loss and set loss. • Epochs = 50 • Batch size = 1	• Training time = 3 hours. • Training Acc = 0.833 • Testing Acc = 0.515 • Not much improvement in accuracy compared to exp 1. • Generates slightly more sharp images.
3)	• Trained scale 1 & 2. • Used mean squared loss and set loss. • Epochs = 5 • Batch size = 1	• Training Acc = 0.484 • Testing Acc = 0.394 • Training was stable • Some loss fluctuations.
4)	• Trained scale 1 & 2. • Used root mean squared loss and set loss. • Epochs = 5 • Batch size = 1	• Training Acc = 0.0 • Testing Acc = 0.0 • Training completely diverged. • Heavy loss fluctuations.
5)	• Trained scale 1 & 2. • Used mean squared loss and set loss. • Epochs = 5 • Batch size = 4	• Training Acc = 0.373 • Testing Acc = 0.301 • Training diverged. • Heavy loss fluctuations.
6)	• Trained scale 1 & 2. • Used mean squared loss and set loss. • Epochs = 5 • Batch size = 1 • Without skip layers	• Training Acc = 0.382 • Testing Acc = 0.362 • Training converged slowly.

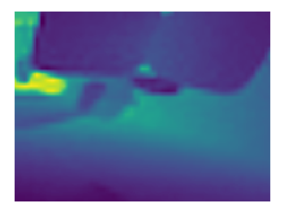
HYPERPARAMETERS TUNED

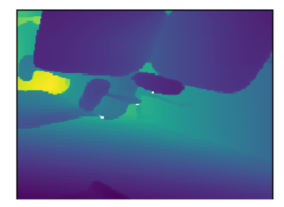
- L2 Weight decay
- Learning rate for each layer
- Batch size
- Set loss regularization
- Learning rate decay
- Gradient clipping range

INSIGHTS

- Network learns slower if the skip layers 1.1 and 1.2 are removed
- After 50 epochs, we could see the depth maps from scale 3 were slightly more detailed when compared to scale 2 results
- The training easily converged to all zero predictions if the learning rate is large
- When a large regularizer is used, the loss stabilizes and doesn't converge
- Downscaling the images did not affect the end predictions significantly
- By overfitting the training to just 3 images, we could get pretty good predictions and from that we inferred our overall pipeline was working


RGB Image


Predicted depth Image


Original depth Image

- Since, the training was performed on just 4000 images, it tends to overfit, and the generalization gap is large, with the test accuracy nearing only 50-51%. However this value is significantly better than what it was without regularization.

CHALLENGES

- Inconsistencies in network design with respect to the paper
- Implementation of set loss
- The original NYU depth dataset was too large to work with (~ 220k images)
- Creating the dataset: Augmenting images from RMRC dataset using multiple transformations
- The hyperparameters mentioned in the paper didn't work and we spent a lot of time tuning them
- Training the network: With the resolution used in the paper(232*310), we faced memory errors in CUDA. We had to scale the images down.

FUTURE WORK

- Use local gradient estimates to enhance current depth predictions. The refined depth maps would minimize difference between estimated depths and estimated gradients.
- Project the results in 3D and evaluate.

REFERENCES

[1] Learning Fine-Scaled Depth Maps from Single RGB Images: https://arxiv.org/pdf/1607.00730.pdf
[2] Depth Map Prediction from a Single Image using a Multi-Scale Deep Network: https://arxiv.org/pdf/1406.2283.pdf
[3] RMRC indoor depth challenge dataset: http://cs.nyu.edu/ silberman/rmrc2014/indoor.php
[4] NYU depth dataset v2: http://cs.nyu.edu/ silberman/datasets/nyu depth v2.html