Monocular Depth Estimation Based on Convolutional Neural Networks

Group: We Want a GPU
Dawai Wang, Ruiyi Wang, Siyi Chen, Yiyang Qiu
{wdwdawei ruiyiw siyich yiyangq}@umich.edu

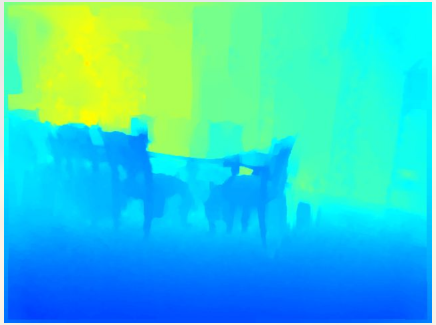
Problem Statement

Problem: Monocular depth estimation

- Predict depth of pixels in a single 2D RGB image

Applications:

- Navigation in robotics, 3D scene reconstruction, augmented reality, etc.



Related Work

DenseDepth[1]

- Used transfer learning on an encoder-decoder framework
- Initialized the encoder with pre-trained denseNet-169

FastDepth[2]

- Applied lightweight model architecture and prune the network
- Achieves similar accuracy, but faster: good for real-time inference

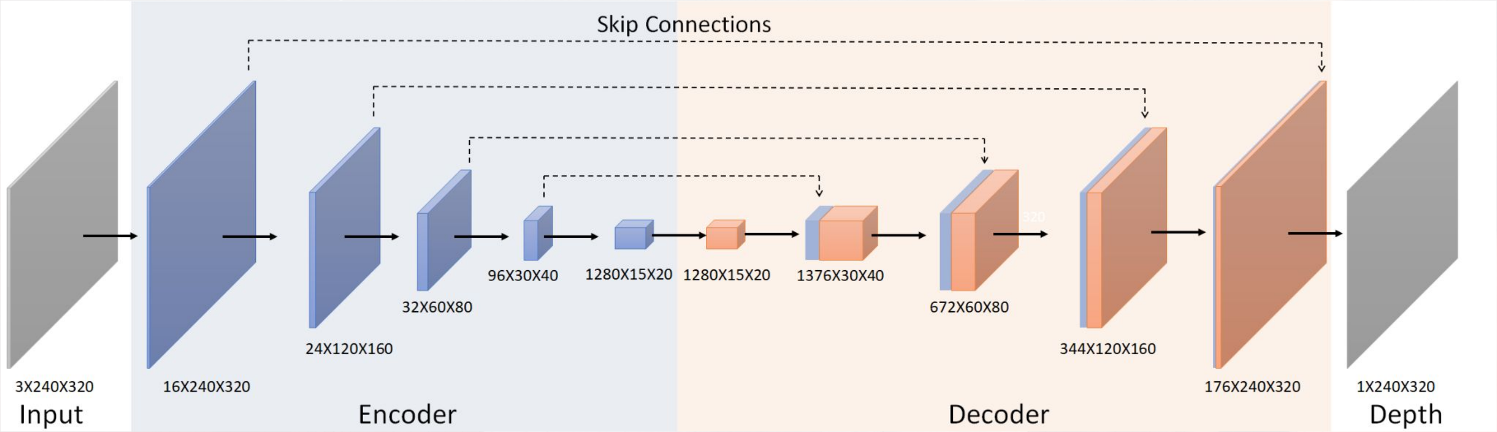
Deep Ordinal Regression[3]

- Applied an SID strategy to discretize depth and utilized ordinal regression
- Avoided complicating network training and reducing resolution of images

Overview

- U-net structure
- Loss function
- Evaluation metrics
- Visualization of results

Model Architecture



- Encoder with transfer learning: mobile net v2/squeeze net
- Decoder with skip connections

Loss Function

$$L(y,\hat{y}) = \lambda L_{depth}(y,\hat{y}) + L_{grad}(y,\hat{y}) + L_{SSIM}(y,\hat{y})$$

$$L_{\text{depth}}(y,\hat{y}) = \frac{1}{n}\sum_{p}^{n} |y_p - \hat{y}_p| .$$

$$L_{grad}(y,\hat{y}) = \frac{1}{n}\sum_{p}^{n} |\boldsymbol{g}_{\mathbf{x}}(y_p,\hat{y}_p)| + |\boldsymbol{g}_{\mathbf{y}}(y_p,\hat{y}_p)| .$$

$$L_{SSIM}(y,\hat{y}) = \frac{1 - SSIM(y,\hat{y})}{2}$$

Experiments & Metrics

1. Encoder-Decoder Structure
   - Mobile net v2
   - Squeeze net

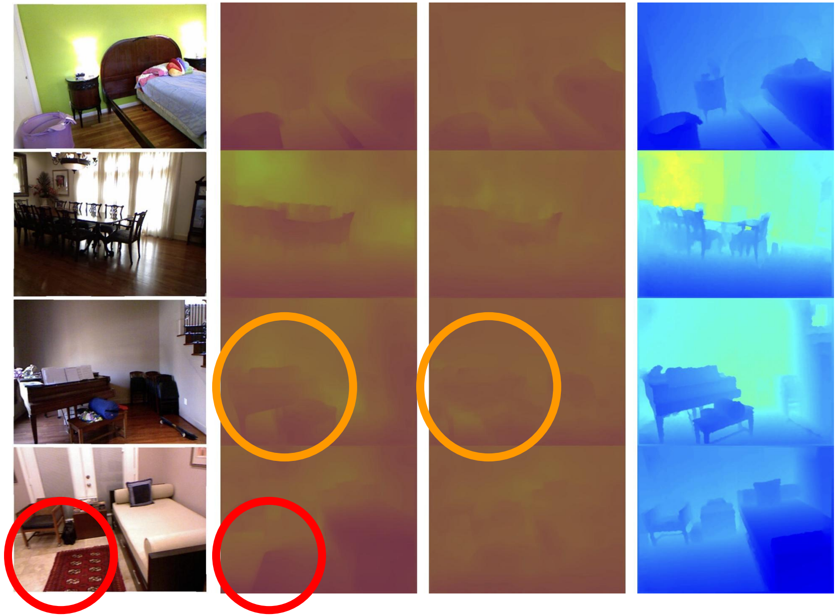
2. Data processing
   - Normalization or not
   - Randomly Gray scaling

1. Average relative error(rel): $\frac{1}{n}\sum_{p}^{n}\frac{|y_p - \hat{y_p}|}{y}$
2. Root mean squared error(rms): $\sqrt{\frac{1}{n}\sum_{p}^{n}(\frac{y_p - \hat{y_p}}{y_p})^2}$
3. Average log10 error: $\frac{1}{n}\sum|\log_{10}(y_p) - \log_{10}(\hat{y_p})|$
4. Threshold accuracy: % of pixels s.t: $\max(\frac{y_p}{\hat{y_p}}, \frac{\hat{y_p}}{y_p}) = \delta < thr$ for $thr = 1.25, 1.25^2, 1.25^3$

Results

	Rel	RMS	$\log_{10}$
Mobile net v2	0.228	0.337	0.088
Squeeze net	0.412	0.575	0.151
	δ_1	δ_2	δ_3
Mobile net v2	0.661	0.891	0.968
Squeeze net	0.405	0.690	0.867

Table 1. Mobile net v2 versus Squeeze net



Input images, Mobile net v2, Squeeze, Ground truth

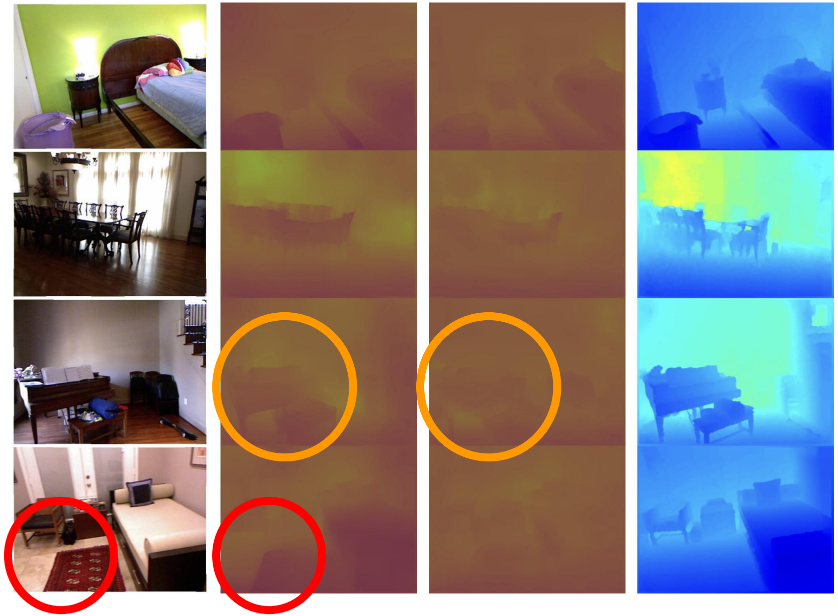
Results

	Rel	RMS	$\log_{10}$
Mobile net v2	0.228	0.337	0.088
Squeeze net	0.412	0.575	0.151
	δ_1	δ_2	δ_3
Mobile net v2	0.661	0.891	0.968
Squeeze net	0.405	0.690	0.867

Table 1. Mobile net v2 versus Squeeze net

	Rel	RMS	log_{10}	δ_1
Normalization	0.3175	0.4234	0.1380	0.4176
Grayscale	0.3195	0.4204	0.1379	0.3936
None	0.3501	0.4732	0.1474	0.3974
	δ_2	δ_3	Loss	
Normalization	0.7306	0.9214	45.3805	
Grayscale	0.7364	0.9258	45.5978	
None	0.6989	0.8924	47.1242	

Table 2. Effect of Normalization and Grayscaling



Input images, Mobile net v2, Squeeze, Ground truth

UNIVERSITY OF
MICHIGAN

Thank you!