fuDeepOrdinalRegression2018: Deep Ordinal Regression Network for Monocular Depth Estimation

Haocheng Zhao

September 16, 2021

Paper

- title: Deep Ordinal Regression Network for Monocular Depth Estimation
- author: Huang Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, Dacheng Tao
- year: 2018 CVPR
- explanation:
  - Monocular Depth Estimation
  - Ordinal Regression

Paper

Research Background

- Monocular Depth Estimation (MDE) progress is slow, comparing to Stereo images or video sequences. A single 2D image may be produced from an infinite number of distinct 3D scenes.
- To overcome this inherent ambiguity, typical methods resort to exploiting statistically meaningful monocular cues or features, such as perspective and texture information, object sizes, object locations, and occlusions.
- Using DCNN-based models improved the MDE performance. These methods address the MDE problem by learning a DCNN to estimate the continuous depth map. This problem is a common regression problem, whose MSE in log-space or its variants are usually adopted as the loss function.

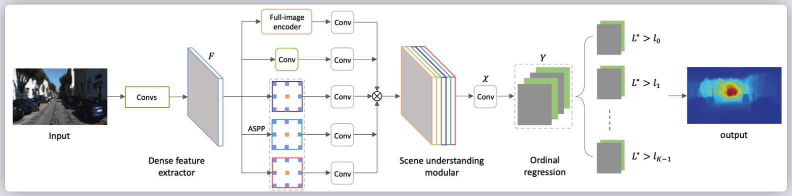
Research Gap

Main Problem

- Image-level information and hierarchical features from deep convolutional neural networks (DCNNs). Map the depth estimation to regression problem and training to minimize mean squared error.
- Existing depth estimation networks employ repeated spatial pooling operations, resulting in undesirable low-resolution feature maps.
- To obtain high-resolution maps, skip-connections or multi-layer deconvolution networks are required. But it complicates network training and consumes much more computations.

Algorithm Architecture

Overall Architecture



two parts:

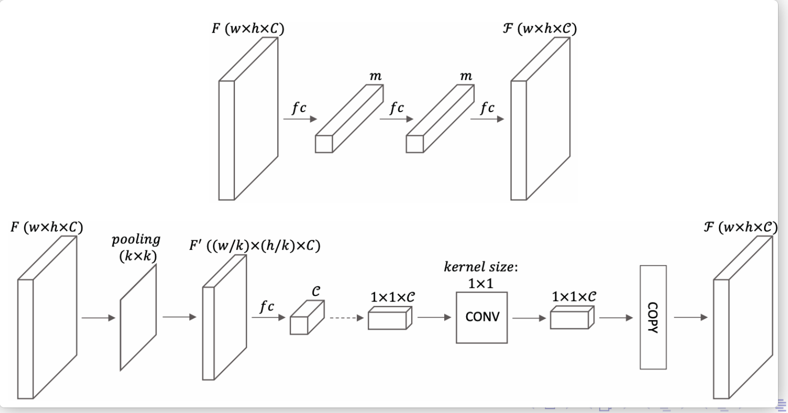
- a dense feature extractor and scene understanding modular
- outputs multichannel dense ordinal labels

Dense Feature Extractor

- Based on some recent Scene Parsing Network, they advocate removing the last few downsampling operators of DCNNs and inserting holes to filters in the subsequent *conv* layers, called dilated convolution, to enlarge the field-of-view of filters without decreasing spatial resolution or increasing the number of parameters.

Scene Understanding Modular

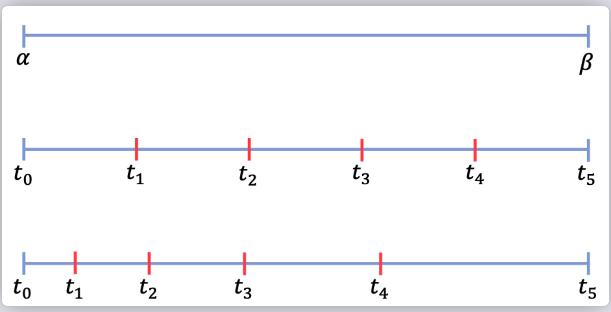
- an atrous spatial pyramid pooling module
- a cross-channel learner 1x1 conv
- a full-image encoder



Ordinal Regression

Spacing-Increasing Discretization and Uniformed

- UD: $t_i = \alpha + (\beta - \alpha) * i/K,$
- SID: $t_i = e^{\log(\alpha) + \frac{\log(\beta/\alpha)*i}{K}},$



Training

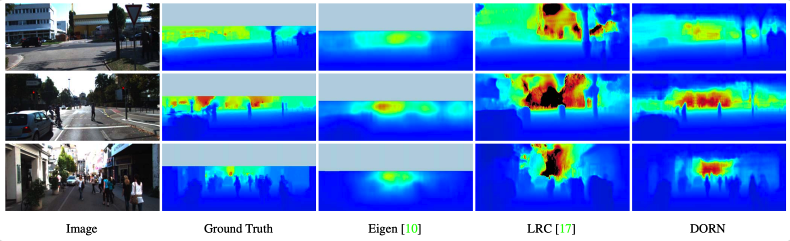
Loss Function

$$
\begin{aligned}
\chi &= \varphi(I, \Phi) \quad Y = \psi(\chi, \Theta) \\
\mathcal{L}(\chi, \Theta) &= -\frac{1}{\mathcal{N}} \sum_{w=0}^{W-1} \sum_{h=0}^{H-1} \Psi(w, h, \chi, \Theta) \\
\Psi(h, w, \chi, \Theta) &= \sum_{k=0}^{l_{(w,h)}-1} \log\left(\mathcal{P}_{(w,h)}^{k}\right) + \sum_{k=l}^{K-1} \left(1 - \log\left(\mathcal{P}_{(w,h)}^{k}\right)\right) \\
\mathcal{P}_{(w,h)}^{k} &= P\left(\hat{l}_{(w,h)} > k \mid \chi, \Theta\right) \\
\mathcal{P}_{(w,h)}^{k} &= \frac{e^{y_{(w,h,2k+1)}}}{e^{y_{(w,h,2k)}} + e^{y_{(w,h,2k+1)}}}
\end{aligned}
\tag{1}
$$

Experiments Design

- Datasets: KITTI [1], Make3D [2, 3], NYU Depth v2 [4], ImageNet [5] for pre-training
- Depth Estimation network based on the Caffe
- Feature Extractor: VGG-16 [6], ResNet-101 [7]

KITTI - Results



KITTI

Method	cap	higher is better			lower is better			
		$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$	Abs Rel	Squa Rel	RMSE	RMSE_{log}
Make3D [49]	0 - 80 m	0.601	0.820	0.926	0.280	3.012	8.734	0.361
Eigen *et al.* [10]	0 - 80 m	0.692	0.899	0.967	0.190	1.515	7.156	0.270
Liu *et al.* [38]	0 - 80 m	0.647	0.882	0.961	0.217	1.841	6.986	0.289
LRC (CS + K) [17]	0 - 80 m	0.861	0.949	0.976	0.114	0.898	4.935	0.206
Kuznietsov *et al.* [31]	0 - 80 m	0.862	0.960	0.986	0.113	0.741	4.621	0.189
DORN (VGG)	0 - 80 m	0.915	0.980	0.993	0.081	0.376	3.056	0.132
DORN (ResNet)	0 - 80 m	**0.932**	**0.984**	**0.994**	**0.072**	**0.307**	**2.727**	**0.120**
Garg *et al.* [15]	0 - 50 m	0.740	0.904	0.962	0.169	1.080	5.104	0.273
LRC (CS + K) [17]	0 - 50 m	0.873	0.954	0.979	0.108	0.657	3.729	0.194
Kuznietsov *et al.* [31]	0 - 50 m	0.875	0.964	0.988	0.108	0.595	3.518	0.179
DORN (VGG)	0 - 50 m	0.920	0.982	0.994	0.079	0.324	2.517	0.128
DORN (ResNet)	0 - 50 m	**0.936**	**0.985**	**0.995**	**0.071**	**0.268**	**2.271**	**0.116**

Make3D - Results

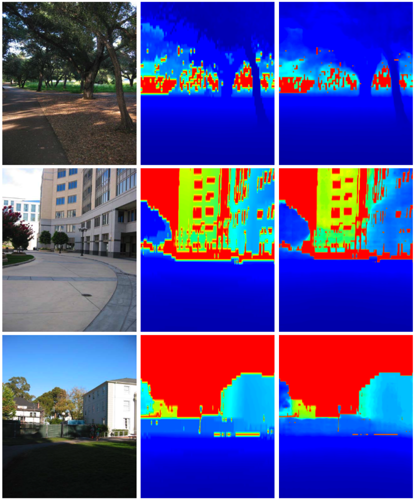


Image Ground Truth DORN

Make3D

Method	C1 error			C2 error		
	rel	$\log_{10}$	rms	rel	$\log_{10}$	rms
Make3D [49]	-	-	-	0.370	0.187	-
Liu *et al.* [37]	-	-	-	0.379	0.148	-
DepthTransfer [26]	0.355	0.127	9.20	0.361	0.148	15.10
Liu *et al.* [39]	0.335	0.137	9.49	0.338	0.134	12.60
Li *et al.* [34]	0.278	0.092	7.12	0.279	0.102	10.27
Liu *et al.* [38]	0.287	0.109	7.36	0.287	0.122	14.09
Roy *et al.* [46]	-	-	-	0.260	0.119	12.40
Laina *et al.* [33]	0.176	0.072	4.46	-	-	-
LRC-Deep3D [57]	1.000	2.527	19.11	-	-	-
LRC [17]	0.443	0.156	11.513	-	-	-
Kuznietsov *et al.* [31]	0.421	0.190	8.24	-	-	-
MS-CRF [58]	0.184	0.065	4.38	0.198	-	8.56
DORN (VGG)	0.236	0.082	7.02	0.238	0.087	10.01
DORN (ResNet)	**0.157**	**0.062**	**3.97**	**0.162**	**0.067**	**7.32**

Figure: c1: 0~80m c2: 0~70m

NYU Depth v2

Method	δ_1	δ_2	δ_3	rel	$\log_{10}$	rms
Make3D [49]	0.447	0.745	0.897	0.349	-	1.214
DepthTransfer [26]	-	-	-	0.35	0.131	1.2
Liu *et al.* [39]	-	-	-	0.335	0.127	1.06
Ladicky *et al.* [32]	0.542	0.829	0.941	-	-	-
Li *et al.* [34]	0.621	0.886	0.968	0.232	0.094	0.821
Wang *et al.* [55]	0.605	0.890	0.970	0.220	-	0.824
Roy *et al.* [46]	-	-	-	0.187	-	0.744
Liu *et al.* [38]	0.650	0.906	0.976	0.213	0.087	0.759
Eigen *et al.* [9]	0.769	0.950	0.988	0.158	-	0.641
Chakrabarti *et al.* [2]	0.806	0.958	0.987	0.149	-	0.620
Laina *et al.* [33]	0.629	0.889	0.971	0.194	0.083	0.790
Li *et al.* [35]	0.789	0.955	0.988	0.152	0.064	0.611
Laina *et al.* [33]†	0.811	0.953	0.988	0.127	0.055	0.573
Li *et al.* [35]†	0.788	0.958	0.991	0.143	0.063	0.635
MS-CRF [58]†	0.811	0.954	0.987	0.121	0.052	0.586
DORN†	**0.828**	**0.965**	**0.992**	**0.115**	**0.051**	**0.509**

Main Evaluation Methods

- threshold δ [8]

$$\max\left(\frac{\hat{d}_p}{d_p}, \frac{d_p}{\hat{d}_p}\right) = \delta < th \tag{2}$$

- Abs Rel, Sq Rel, RMSE and $\text{RMSE}_{\log}$

$$\text{abs rel.} = \frac{1}{n}\sum\left|\frac{y_{pred} - y_{gt}}{y_{gt}}\right| \tag{3}$$

$$\text{sq. rel.} = \frac{1}{n}\sum\left(\frac{y_{\text{pred}} - y_{gt}}{y_{gt}}\right)^2 \tag{4}$$

$$RMSE = \sqrt{\frac{1}{n}\sum\left(y_{pred} - y_{gt}\right)^2} \tag{5}$$

$$\log RMSE = \sqrt{\frac{1}{n}\sum\left(\log\left(y_{pred}\right) - \log\left(y_{gt}\right)\right)^2} \tag{6}$$

- A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, Sep. 2013.
- A. Saxena, M. Sun, and A. Y. Ng, "Make3D: Learning 3D Scene Structure from a Single Still Image," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 5, pp. 824–840, May 2009.
- A. Saxena, S. Chung, and A. Ng, "Learning Depth from Single Monocular Images," in *Advances in Neural Information Processing Systems*, vol. 18. MIT Press, 2006.
- N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor Segmentation and Support Inference from RGBD Images," in *Computer Vision – ECCV 2012*, ser. Lecture Notes in Computer Science, A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, Eds. Berlin, Heidelberg: Springer, 2012, pp. 746–760.

- O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *Int J Comput Vis*, vol. 115, no. 3, pp. 211–252, Dec. 2015.
- K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *arXiv:1409.1556 [cs]*, Apr. 2015.
- K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," *arXiv:1512.03385 [cs]*, Dec. 2015.
- L. Ladicky, J. Shi, and M. Pollefeys, "Pulling Things out of Perspective," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*. Columbus, OH, USA: IEEE, Jun. 2014, pp. 89–96.