Accurate Monocular Depth Estimation via Interaction of Hierarchical Features

Jiahao Zhang1a, Dan Chen1*, Ye Lin1 and You Wu1

1School of Electrical Engineering and Automation, Fuzhou University, Fuzhou, Fujian, 350100, China
2n180120080@fzu.edu.cn *Corresponding author’s e-mail: fzuchd@163.com

Abstract. Monocular depth estimation is a challenging task, which assists in understanding 3D scene geometry from the same 2D scene. The ordinal-regression-method demonstrates superior performance in this issue but naive ordinal inference strategy for inferring the final depth values and naive operations to up-sample to the desired space scale introduce significant discretization errors and object boundary confusion. Firstly, we come up with a novel inference strategy to reduce the discretization errors. And then a specifically designed decoder that completes the fusion of different hierarchical features under guidance and the fusion feature reconstruction. We evaluate on a public monocular depth-estimation benchmark dataset (NYU Depth V2). The experimental results show that the method proposed outperforms other ordinal regression methods.

1. Introduction

Depth estimation from 2D images is a challenge but a critical task in computer vision, which is applied to scene understanding, 3D reconstruction, segmentation and autonomous driving. However, it is an ill-posed problem to estimate the depth of the scene from a single 2D image since infinitely many 3D scenes can be projected to the same 2D scene. With the advent of deep learning, the Monocular Depth Estimation (MDE) has witness significant progress[2-14], and most of the previous works adopted regression methods to estimate depth values, which introduce the ambiguity of regressing to a true depth.

Inspired from Fu. et al[11], who proposed an ordinal regression method called DORN, regard the MDE to a simple but specific task which only judges whether the estimated depth larger than the corresponding label depth discretization. In addition, the DORN outperformed other MDE methods by a significant margin. However, the inference strategy for inferring the continuous depth values is lack of smoothness and boundary constraints and has severe grid artifacts. What’s more, naive up-sampling to original scale or concatenate with low hierarchical features without guidance contributing to the boundary blurred.

2. Related Work

Eigen, et al. firstly estimated the monocular depth by introducing deep learning and significantly promoted the accuracy and feasibility of the MDE issue. Recently, more and more methods based on deep learning are proposed to continue improving the metrics of the MDE, and the main comparison method[11] of this paper firstly regarded the MDE as an ordinal regression problem and acquires the SOTA. However, existing methods focus on the pixel-wise accuracy and high accuracy of metrics but ignore qualitative aspects like depth consistency and edge accuracy. To the above issue, we will briefly outline two related sets of work that are relevant to this paper, namely, supervised depth estimation and attention mechanism.
In monocular depth estimation, supervised approaches take a pair of RGB-D images as the input. Most of the works designed a decoder or more effective modules to capture the depth information of RGB images and then generate depth images to compute loss with ground truth for supervision in training. [6,12] integrate the random field theory into CNNs to realize local optimization, Qi, et al.[10] investigate the duality between depth map and surface normal. Jiao, et al[13] proposed that combined object segmentation with depth estimation and Lee, et al[14] suggested that estimating the relative depth between pairs of pixels by using DNNs. Fu, et al[11] formulated the MDE to ordinal regression and achieved state-of-the-art.

In computer vision, the attention mechanism has been widely used in pixel-level issues and verified the powerful ability to capture long-range dependency. Nonlocal[16] first adopts self-attention mechanism as a module for computer vision tasks, such as video classification, object detection and instance segmentation. Hu, et al. [17] proposed a simple way named SE block to model global context features and Cao, et al incorporated the non-local into SE block computes the attention map of every pixel without introducing higher computation. Inspired by Cao's work, we assume each feature was treated that only reflects one relationship of depth, but the different features may contain the same or parallel of the relationship of depth. Base on the above, a simple attention unit named FRU is proposed to capture non-local information and re-construct the features for better qualitative performance.

3. Proposed Methods

3.1. Architecture overview
As the Figure 1. shown the detailed architecture of the proposed model, it can be conceptually divided into two main parts.

Figure 1. Structure of overall network. Red dash lines flow the features whose space scale are \( \frac{1}{2} \) of the input size, green ones flow the features whose space scale are \( \frac{1}{4} \) of the input size and purple ones flow the features whose space scale are \( \frac{1}{8} \) of the input size.

3.1.1. Encoder
Because of the ill-posed for MDE, many outperform feature extraction networks like VGGNet[19], ResNet[20] and DenseNet[21] are utilized to obtain high degree scene comprehension. In this paper, both ResNet-50 and ResNet-101 are considered as the encoder due to the trade-off of the amounts of parameters and performance.

3.1.2. Decoder
The remaining parts of the model all belong to the decoder and structure as Figure 2 demonstrated. This is the sub-network for decoding the features which are abundant of depth clues and then they fused with the shallow features to fined the result of depth estimation. It consists of a series of \( 1 \times 1 \) convolutions, Multi-scale Context Fusion Block (MCFB) and Features Reconstruction Unit (FRU) respectively. All convolutions used to condense the features and project to the demanded dimension. MCFB concatenates the different hierarchical features whose sizes have been scaled by up- or down- operations, and then they are fused under the guidance of the other branches’ front features. The middle and up branch of FRU re-mines the features that contain the same or parallel of the relationship of depth and models the relationship weights, then, element-wise multiplies with bottom branch’s output to activate the depth similarity of every two pixels and finally utilizes a residual connection to prevent vanishing gradient
problem.

Figure 2. Structure of decoder. Downs mean stride convolution and ups mean bi-linear interpolate sample. And C means concatenation, σ means sigmoid function, · means inner product, × means element-wise multiply, + means addition and S means softmax function. The part enclosed by the blue rectangle is MCFB, that enclosed by the gray rectangle is FRU, both of them compose the decoder.

3.2. Depth Discretization
In our work, MDE is considered as an ordinal regression problem, so the true depth values are discretized into multiple classes, where each class corresponds to a depth interval. Following the \cite{[11]}, the spacing-increasing discretization (SID) is adopted to acquire the depth interval. The SID can be formulated as: $t_k = e^{\text{ind}_k \cdot \text{ind}_{\text{max}} / k \cdot \text{ind}_{\text{min}} \times k}$ mathematically, where $[d_{\text{min}}, d_{\text{max}}]$ is the depth range of the given training dataset, $k$ is the discretized label of the given pixel and $k \in [0, K - 1]$, above parameters all used to compute the discretization threshold $t_k$. In this paper, $K$ is set 80 to balance the performance and the numbers of network parameters.

3.3. Ordinal Regression
Different from the regression-based-method MDE, the output of the proposed network is classified probability map of every pixel, not the final depth maps. The classified probability map is computed by $K$ binary classifier where the $k$th predictor only learn to predict whether the given pixel greater than the depth value belonging the discretization threshold $t_k^k$. To train the classifier, a $K$ size rank vector is created to judge the given pixel belonging to $[t_k, t_{k+1}]$, if true the value set 1 else set 0. The classified probability map contains $2 \times K$ predict results of every pixel because of every two predict results corresponding a binary classifier. Therefore, the pixel-wise ordinal regression loss formulated as:

$$L_i = \sum_{k=0}^{K-1} \ln(p_i^k) + \sum_{l_i}^{K-1} \ln(1 - p_i^k),$$

where $i$ is the index of the pixel, $p_i^k$ is computed by softmax function on the binary classifier results and $l_i'$ is the discretized depth label.

3.4. Inference Strategy
The confidence maps $p_i^k$ that are used to inferred the final depth maps are reflected the depth information of every pixel. \cite{[16]} adopted hard inference strategy to acquire depth maps which results in sever grid artifacts and unsmooth depth transition. A novel inference strategy is proposed to alleviate the above phenomena. It formulated as:

$$d_i = \frac{t_i^{l_i+1} + t_i^l}{2} \times (1 - f_i) + \frac{t_i^{l_i+1} + t_i^{l_i+2}}{2} \times f_i + \frac{2}{K}, l_i = \left\lfloor \sum_{k=0}^{K-1} p_i^k \right\rfloor, f_i = \sum_{k=0}^{K-1} \frac{p_i^k}{l_i}, l_i$$
where \( \lfloor \cdot \rfloor \) is the floor operation. The neighbour discretized labels of the given pixel both used to calculate the depth for ensuring varying stably and the estimated depth classes are summed by all probabilities of all ranks instead of counting the numbers of probabilities whether greater than 0.5.

4. Experiments and Analysis

4.1. Dataset: NYU depth v2 and Metrics

The original NYU depth V2[1] dataset consists of 240K RGB-D images having size of 480×640 acquired as video sequences using a Microsoft Kinect camera from 464 indoor scenes. Following the official train/test split as previous works, where 249 scenes for training and 215 for testing, and testing the model by using the official 654 images. Among the entire NYU depth V2 dataset, we sample approximately 12K unique training RGB-D pairs with a fixed sampling frequency. For training, all images are resized to 288×384 and then randomly crop to 256×352 pixels, as input to the network. Following the previous works[2], online data augmentation strategies are adopted to increase the diversity of samples, which include random scaling, random rotation, color, flips, and contrast. For evaluation, we use following metrics used by previous works:

\[
\text{Threshold} \% \text{ of } d_i \text{ s.t. max} \left( \frac{d_i}{d^*_i} \right) = \delta < \text{thr}, \text{thr} = 1.25^1, 1.25^2, 1.25^3, \text{rmse}: \sqrt{\frac{1}{N} \sum_i \left( d_i - d^*_i \right)^2}, \text{rmse}^{\log}: \sqrt{\frac{1}{N} \sum_i \left( \log d_i - \log d^*_i \right)^2}, \text{rel}: \frac{1}{N} \sum_i \left| \frac{d_i - d^*_i}{d^*_i} \right|, \text{log10}: \frac{1}{N} \sum_i \left| \log_{10} d_i - \log_{10} d^*_i \right|,
\]

where \( N \) denotes a collection of pixels that ground truth values are available, \( d_i \) and \( d^*_i \) are estimation and the ground truth value, respectively.

4.2. Implement Details

We implement the proposed network using the open deep learning framework Pytorch on a single Nvidia GTX1080Ti GPU. For training, we used SGD optimizer with base learning rate \( 1 \times 10^{-4} \), where momentum and weight decay are set to 0.9 and \( 5 \times 10^{-4} \). The total number of epochs is set 50 with batch size 8 for all experiment in this work. As the candidates of the encoder, both Resnet-50 and Resnet-101 are utilized, whose parameters are pretrained on the ImageNet classification task[15].

4.3. Ablation Study

We conduct evaluations with different inference strategy versions to see the effectiveness of the proposed strategy as well as evaluated without FRU to verify its reconstruction ability of features. From the base model which only consists of backbone network, a series of convolutions and MCFB, we add the base model with FRU and then change the inference strategy to see how the added factor improves accuracy and the result is given in Table.1. As the FRU join in, the overall performance is improved, moreover, our proposed inference strategy continues to improve various metrics.

4.4. Result and Discussion

As Table.2 illustrates, most of the metrics outperform the recent works and Figure.2 also demonstrates the quantitative results that have accurate boundaries and smooth depth variations. The accurate boundaries can be attributed to the MCFB fusing the hierarchical features and make up the lack of high-level features. And the FRU reconstruct the feature for representing the depth clues better and active the relation of every pair of pixels.

5. Conclusion

In this paper, we have presented a supervised monocular depth estimation network and achieve better results compared with recent SOTA methods with less training data and without abstruse training tricks. By rethinking the concatenation operations of different hierarchical features, convolutions and MCFB are adopted to conquer the gaps of different level features so that more details and preciser geometric consistency are easier to acquire. And a specific attention module FRU is utilized, more depth clues in
final generated features could be used to support the final depth estimation. And a novel ordinal inference strategy was brought up to reduce the error and generate smoother depth maps. The experiments on the NYU v2 dataset well demonstrate the superiority of our model.

![Image](https://example.com/image1)

Figure 3. Qualitative comparison on NYU Depth V2 dataset. From left to right, Image, Label, Prediction, Fu et al.\textsuperscript{[11]} and ours.

| Variants            | \(\text{rel}_\downarrow\) | \(\text{rmse}_\downarrow\) | \(\text{rmse}_{\text{log}}\) | \(\delta_1\) \(\uparrow\) | \(\delta_2\) \(\uparrow\) | \(\delta_3\) \(\uparrow\) |
|---------------------|----------------|----------------|----------------|----------------|----------------|----------------|
| Base(hard)          | 0.140          | 0.505          | 0.176          | 0.059          | 0.822          | 0.965          | 0.990          |
| Base+FRU(hard)      | 0.137          | 0.495          | 0.173          | 0.058          | 0.826          | 0.965          | 0.991          |
| Base+FRU(ours)      | 0.133          | 0.489          | 0.171          | 0.056          | 0.827          | 0.967          | 0.992          |

Table 1. Comparisons of different variants on the NYU depth v2 with ResNet-50 backbone.

|                          | \(\text{rel}_\downarrow\) | \(\text{rmse}_\downarrow\) | \(\delta_1\) \(\uparrow\) | \(\delta_2\) \(\uparrow\) | \(\delta_3\) \(\uparrow\) |
|--------------------------|----------------|----------------|----------------|----------------|----------------|
| Saxena et al. [3]        | 0.349          | 1.214          | 0.447          | 0.745          | 0.897          |
| Chakrabarti et al. [9]   | 0.149          | 0.620          | 0.806          | 0.958          | 0.987          |
| Eigen et al. [2]         | 0.121          | 0.641          | 0.769          | 0.950          | 0.988          |
| Ren et al. [4]           | 0.113          | 0.501          | 0.833          | 0.968          | 0.993          |
| Laina et al.(R50) [5]    | 0.127          | 0.573          | 0.811          | 0.953          | 0.988          |
| Li et al.(R50) [8]       | 0.147          | 0.601          | 0.808          | 0.957          | 0.985          |
| Qi et al.(R50) [10]      | 0.128          | 0.569          | 0.834          | 0.960          | 0.990          |
| Xu et al.(R50) [6]       | 0.121          | 0.586          | 0.811          | 0.954          | 0.987          |
| Zhang et al.(R50) [7]    | 0.121          | 0.497          | 0.846          | 0.968          | **0.994**      |
| Li et al.(R101) [8]      | 0.139          | 0.545          | 0.820          | 0.960          | 0.989          |
| Fu et al.(R101) [11]     | 0.115          | 0.509          | 0.828          | 0.965          | 0.992          |
| Ours(R50)                | 0.133          | 0.489          | 0.827          | 0.967          | 0.992          |
| Ours(R101)               | 0.125          | **0.465**      | **0.848**      | **0.970**      | **0.994**      |

Table 2. Quantitative comparison with recent state-of-the-art on NYU depth v2

Acknowledgments
Dan Chen is supported by the Natural Science Foundation of Fujian Province of China (Grant no. 2018J01534) and Scientific Research Project of Fuzhou University Science and Education Park Development Center in Jinjiang City. (Grant no. 2019-JJFDKY-55).

References
[1] Siberman N, Hoiem D, Kohli P, et.al. (2012) Indoor segmentation and support inference from rgbd images[C]. In: ECCV. Springer, Berlin, Heidelberg. pp 746-760
[2] Eigen D, Fergus R. (2015) Predicting depth, surface normals and semantic labels with a common mutil-scale convolutional architecture[C]. In: CVPR. Boston. pp 2650-2658
[3] A. Saxena, M. Sun, and A. Y. Ng. (2009) Make3d: Learning 3d scene structure from a single still image. IEEE transactions on pattern analysis and machine intelligence, 31(5):824–840

[4] Ren, H., El-khamy, M., Lee, J. (2019) Deep robust single image depth estimation neural network using scene understanding. https://arxiv.org/pdf/1906.03279.pdf

[5] Laina I, Rupprecht C, Belagiannis V, et.al. (2016) Deeper depth prediction with fully convolutional residual networks[C]. 3D Vision(3DV), Fourth International Conference on IEEE, 2024-2039

[6] Xu D, Ricci E, Ouyang W, et.al. (2018) Monocular depth estimation using multi-scale continuous crfs as sequential deep networks. https://arxiv.org/pdf/1803.00891.pdf

[7] Zhang, Z., Cui, Z., Xu, C., Yan, Y., Sebe, N., Yang, J. (2019): Pattern-affinitive propagation across depth, surface normal and semantic segmentation. In: CVPR. California Long Beach. pp. 4106–4115

[8] Bo Li, Yuchao Dai, and Mingyi He. (2017) Monocular depth estimation with hierarchical fusion of dilated cnns and soft-weighted-sum inference. https://arxiv.org/pdf/1708.02287.pdf

[9] A. Chakrabarti, J. Shao, and G. Shakhnarovich. (2016) Depth from a single image by harmonizing overcomplete local network predictions. In Advances in Neural Information Processing Systems, pp 2658–2666

[10] X. Qi, R. Liao, Z. Liu, R. Urtasun, and J. Jia. (2018) Geonet: Geometric neural network for joint depth and surface normal estimation. In: CVPR. Salt Lake City. pp 283–291

[11] Fu H, Gong M, Wang C, et.al. (2018) Deep Ordinal Regression Network for Monocular Depth Estimation[C]. In: CVPR. Salt Lake City. pp 2002-2011

[12] Fayao Liu, Chunhua Shen, and Guosheng Lin, (2014) Deep convolutional neural fields for depth estimation from a single image, https://arxiv.org/pdf/1411.6387.pdf.

[13] Jiao, J., Cao, Y., Song, Y., Lau, R. (2018) Look deeper into depth: Monocular depth estimation with semantic booster and attention-driven loss. In: ECCV. Munich. pp. 53–69

[14] Lee, J.H., Kim, C.S. (2019) Monocular depth estimation using relative depth maps. In: CVPR. California Long Beach. pp.9729–9738

[15] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, and Michael Bernstein. (2015) Imagenet large scale visual recognition challenge. International Journal of Computer Vision, 115(3):211–252

[16] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. (2018) Non-local neural networks. In: CVPR, Salt Lake City. pp 7794–7803

[17] J. Hu, L. Shen, and G. Sun. (2017) Squeeze-and-Excitation Networks. https://arxiv.org/pdf/1709.01507.pdf

[18] Y. Cao, J. Xu, S. Lin, F, Wei and H. Hu. (2019) GCNet: Non-local Networks Meet Squeeze-Excitation Networks and Beyon. https://arxiv.org/pdf/1904.11492.pdf

[19] K. Simonyan and A. Zisserman. (2014) Very deep convolutional networks for large-scale image recognition. https://arxiv.org/abs/1409.1556

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. (2015) Deep residual learning for image recognition, https://arxiv.org/abs/1512.03385.pdf

[21] H. Gao, L. Zhuang, and et. al. (2020) Convolution Network with Dense Connectivity, https://arxiv.org/abs/2001.02394.pdf