CLIFFNet for Monocular Depth Estimation with Hierarchical Embedding Loss

Lijun Wang[1][0000−0003−2538−8358], Jianming Zhang[2][0000−0002−9954−6294], Yifan Wang[1⋆][0000−0003−1223−136X], Huchuan Lu[1,3][0000−0002−6668−9758], and Xiang Ruan[4][0000−0003−4500−7516]

[1] Dalian University of Technology, China
[2] Adobe Research, USA
[3] Peng Cheng Lab, China
[4] tiwaki Co.,Ltd., Japan
{ljwang,wyfan,lhchuan}@dlut.edu.cn
jianmzha@adobe.com ruanxiang@tiwaki.com

Abstract. This paper proposes a hierarchical loss for monocular depth estimation, which measures the differences between the prediction and ground truth in hierarchical embedding spaces of depth maps. In order to find an appropriate embedding space, we design different architectures for hierarchical embedding generators (HEGs) and explore relevant tasks to train their parameters. Compared to conventional depth losses manually defined on a per-pixel basis, the proposed hierarchical loss can be learned in a data-driven manner. As verified by our experiments, the hierarchical loss even learned without additional labels can capture multi-scale context information, is more robust to local outliers, and thus delivers superior performance. To further improve depth accuracy, a cross level identity feature fusion network (CLIFFNet) is proposed, where low-level features with finer details are refined using more reliable high-level cues. Through end-to-end training, CLIFFNet can learn to select the optimal combinations of low-level and high-level features, leading to more effective cross level feature fusion. When trained using the proposed hierarchical loss, CLIFFNet sets a new state of the art on popular depth estimation benchmarks.

Keywords: Monocular Depth Estimation, Hierarchical Loss, Hierarchical Embedding Space, Feature Fusion.

1 Introduction

Depth estimation is traditionally tackled by shallow models [20, 22] with handcrafted features. More recent works [6, 14] have shown that the success of deep convolutional neural networks (CNNs) in other computer vision areas can also be transferred to monocular depth estimation. The hierarchical structure of trainable CNN features provides stronger representation capabilities, yielding more accurate monocular depth estimation.

⋆ Corresponding author

To train CNNs, a well defined loss function is required in the first place to provide supervision by measuring the differences between predictions and targets. A wide range of loss functions have been explored in the literature of depth estimation. For instance, the reverse Huber loss [14] and depth aware loss [11] are used to address the heavy-tailed distribution of depth values in some existing datasets, while the scale invariant loss [6] and depth gradient loss [18] are designed to balance depth relations and scales. Although good performance has been achieved, these losses are manually designed, which require rich domain knowledge. As such, their generalization ability across different datasets cannot be guaranteed. Besides, the existing depth losses are mostly defined on a per pixel basis, which fail to capture context information. Therefore, they may be over sensitive to label noise and outliers, leading to unstable network training. In order to address the above issues, it is interesting to investigate alternative representation spaces where training losses can be more effective and robust for depth supervision.

In light of the above observations, we propose to leverage a loss function computed in a hierarchical embedding space for training depth estimation models. To this purpose, we devise different hierarchical embedding generators (HEGs) which take depth maps as input and generate their corresponding hierarchical embedding spaces, which in our cases are hierarchical convolutional feature maps extracted from the input depth maps. The loss function for training depth estimation networks is then computed on both the original depth space and the generated embedding space, giving rise to a *hierarchical embedding loss*. In order to seek desired hierarchical embeddings, we design multiple tasks to train HEGs. It is found that training on relevant tasks even without additional annotations can effectively improve depth estimation performance. It can also be shown that the widely adopted gradient loss is a special form of our hierarchical loss computed by a HEG with hand-designed network parameters. However, our experiments confirm that properly trained HEGs can significantly outperform either hand-designed ones or those trained on irrelevant tasks.

Another contribution of this paper is a cross level identity feature fusion (CLIFF) module acting as a basic building block of our depth estimation network. Fully convolutional networks with multi-level feature pyramids have become the *de facto* technique for solving pixel-level prediction tasks [23, 32] A number of evidences [23, 28, 19] suggest that high-level features with more semantic and global context information is able to facilitate more reliable and accurate predictions. In comparison, low-level features with higher resolutions contain more detailed local information, which may benefit high-resolution predictions. Nonetheless, the low-level features also carry more noise which may reduce the reliability of the predictions. In light of the above observations, given features of two different levels the proposed CLIFF module first enhances low-level features using high-level ones through an attention scheme. In addition, the proposed architecture allows our CLIFF module to learn to select optimal features from the combination of high-level, original and enhanced low-level features. Finally, an identity mapping path connecting the high-level input feature

and output is built to preserve the reliable semantic information. By applying CLIFF modules recursively, we obtain a new depth estimation network termed as CLIFFNet.

Our main contribution can be summarized into three folds.

- A new form of hierarchical loss computed in depth embedding spaces is proposed for depth estimation.
- Different architectures and training schemes of hierarchical embedding generators are investigated to find desirable hierarchical losses.
- A new CLIFFNet architecture is designed with more effective cross level feature fusion mechanism.

When trained using the proposed hierarchical losses, our CLIFFNet sets new state-of-the-art performance on popular depth estimation benchmarks.

2 Related Work

Monocular depth estimation is a long standing problem in computer vision [10, 26, 20, 22]. Recent years have witnessed tremendous progress achieved by deep learning based depth estimation methods. In the seminal work by Eigen *et al.* [6], a multi-scale deep network based method is proposed, where a global network is used to predict coarse-scale depth and a local network further refines the prediction with finer details. This network is extended by [5] into three levels, and is successfully applied to depth prediction, normal estimation and segmentation. Later on, Laina *et al.* [14] propose one of the earliest fully convolutional network architectures for monocular depth estimation, which significantly boosts the estimation accuracy. Motivated by [14], convolutional architectures have been intensively studied for depth estimation. For instance, a two-stream convolutional network is proposed in [17], which simultaneously predicts depth and depth gradients to restore fine depth details. Fu *et al.* [7] discretize depth values and propose a deep ordinal regression network. In contrast, [16] decomposes metric depth prediction into relative depth prediction and recombination, where a new convolutional network is proposed for relative depth estimation. Recently, Zhi *et al.* [36] proposes a new type of convolution which condisers the camera parameters to learn calibration-aware patterns for monocular depth estimation. In addition, different training strategies have been explored to benefit monocular depth estimation, including multi-task training [33, 36, 30], self-supervised learning with photometric losses [8, 31], and those with sparse ordinal [2] or relative depth [32, 29] superivsions.

Although, the above deep learning based methods have significantly improved depth prediction accuracy, the scheme of deep feature fusion across levels is not thoroughly studied for depth estimation. Nonetheless, our experiments show that effective multi-level feature fusion can yield considerable performance boost.

Another line of work which correlates to ours is the design of loss functions for training depth estimation networks. Among others, [6] proposes a scale invariant loss, which enforces the network to learn depth relations rather than scales. In a

similar spirit, Li *et al.* [18] propose depth gradient loss, which computes the L1 losses in the gradient space of the predicted and ground truth depth. Meanwhile, the heavy-tailed distribution of depth values have been observed in both [14] and [11]. They propose to address this issue using the reverse Huber loss and depth-aware loss, respectively, both of which attach higher weight towards samples with large residuals. Our hierarchical losses differ from the above works mainly in two aspects. First, most of the above losses are manually designed, whereas ours can be learned in a data-driven manner on relevant tasks. Second, the above losses are mostly defined in the original depth space on a per pixel basis. In comparison, ours are defined in hierarchial embedding spaces of the depth. Our experiments show that the hierarchical loss can capture contextual information and is more robust to local noises, leading to significant performance gain. Our hierarchical loss is also related to perceptual losses [12, 25]. However, the methods in [12, 25] aim to improve visual quality of image generation/reconstruction by directly applying perceptual losses, while our focus is on architecture design and relevant task exploration to achieve more superior hierarchical embeddings to compute hierarchical losses.

3 Method

3.1 Hierarchical Embedding Loss for Depth Estimation

For monocular depth estimation, a deep network takes a single image as input and estimates its depth map $\hat{\boldsymbol{d}}$. Given the corresponding ground truth depth $\boldsymbol{d}$ and a loss function $L(\boldsymbol{d}, \hat{\boldsymbol{d}})$ measuring the differences between the prediction and ground truth, the parameters of the network can then be learned by minimizing the loss function. Instead of directly comparing the differences in the original depth space, some existing works demonstrate that loss functions defined on some manually designed embeddings (*eg.*, vertical and horizontal gradients) of the original depth may embody more appealing properties, leading to considerable accuracy gains. Motivated by this fact, we aim to design an embedding generator $G(\boldsymbol{d}, \boldsymbol{\theta})$ parameterized by $\boldsymbol{\theta}$ to map the input depth into an embedding space. As such, the parameter $\boldsymbol{\theta}$ can be learned in a data-driven manner rather than through hand-engineering.

Inspired by the impressive performance of hierarchical structures in deep networks, we propose to transfer their success to the supervision domain by defining loss functions on hierarchical embedding spaces. To this end, we implement the hierarchical embedding generator (HEG) G using multi-layer CNNs[5]. By feeding a depth map $\boldsymbol{d}$ into G, we obtain a set of K hierarchical convolutional feature maps $\{G_1(\boldsymbol{d}), G_2(\boldsymbol{d}), \ldots, G_K(\boldsymbol{d})\}$, which are treated as an embedding hierarchy of the input depth. The final loss function can then be computed as:

$$L_D(\boldsymbol{d}, \hat{\boldsymbol{d}}) = \sum_{k=0}^{K} w_k L(G_k(\boldsymbol{d}), G_k(\hat{\boldsymbol{d}})), \tag{1}$$

[5] We drop the parameter θ for notational simplicity.

Table 1. Architecture details of HEG-S. Conv, Max, FC, BN, NS, SM, and α denote convolutional layers with kernel size 3×3, adaptive max pooling with output size 2×2, fully connected layer, batch normalization, number of scenes, softmax layer and negative slop of leaky ReLUs, respectively.

#Layer	1	2	3	4	5	6	7	8	9	10
Type	Conv	Conv	Conv	Conv	Conv	Conv	Max+ Flatten	FC	FC	FC
Output Channel	16	16	32	32	64	64	256	256	256	NS
Stride	1×1	1×1	2×2	1×1	2×2	1×1	–	–	–	–
α	0.01	0.01	0.01	0.01	0.01	0.01	–	0.0	0.0	–
Normal.	BN	BN	BN	BN	BN	BN	–	BN	BN	SM

where G_0 denotes the identify mapping, *ie.*, $G_0(\boldsymbol{d}) = \boldsymbol{d}$; w_k indicates the loss weight. As a result, the above loss function combines the supervision from both the original depth space and its embedding spaces of different levels.

A essential problem remaining is how to learn the parameters of HEGs. In this paper, we identify appropriate tasks for training HEGs according to the following two standards.

- The task, including both the input and output target, should be relevant to depth estimation. Otherwise, the learned HEGs can hardly benefit depth estimation. Consider a HEG pre-trained on image classification, which can also be adopted for training depth estimation. However, our experiments show that its performance in terms of depth accuracy gain is similar to a randomly initialized HEG.
- Although additional annotations maybe beneficial, we focus on tasks that require limited additional manual annotations. As a result, the idea of learning hierarchical embedding losses can be more easily applied across different datasets, and the comparison against baseline approaches trained without a hierarchical loss is more fair.

According to the above standards, we mainly select depth-based scene classification and depth reconstruction as two tasks for training HEGs. We further design appropriate HEG network architectures for the two tasks and study their impact on depth estimation.

HEG-S trained on depth-based scene classification The image and depth sample pairs in existing datasets are collected in various locations and scenes. For instance, the NYU-Depth V2 dataset [27] contains 464 scenes, while the data of Cityscape [3] belong to 50 scenes. The scene name of each sample can be easily recorded as meta data when collecting the depth data (*eg.*, in many datasets the depth samples recorded under the same scene are stored in one folder), and therefore does not require heavy manual labour for additional annotations. Motivated by this observation, we propose a depth-based scene classification

Table 2. Architecture details of HEG-R encoder. Conv, Max, FC, and α denote convolutional layers with kernel size 3×3, adaptive max pooling with output size 2×2, fully connected layer, and negative slop of leaky ReLUs, respectively.

#Layer	1	2	3	4	5	6	7	8
Type	Conv	Conv	Conv	Conv	Conv	Conv	Max+ Flatten	FC
Output Channel	16	16	32	32	64	64	256	256
Stride	1×1	1×1	2×2	1×1	2×2	1×1	–	–
α	0.01	0.01	0.01	0.01	0.01	0.01	–	0.0

task to train HEG. Technically, the HEG takes as input a depth map, rather than an RGB image, and is trained to infer its corresponding scene label from a pre-defined label set. It is very likely that depth maps taken from the same scene share similar properties, *eg.*, depth scales and structures. By learning to identify the correlation between depth and scenes, we hope the embeddings generated by the trained HEG are able to capture the key properties of the input depth map, and further benefit depth estimation training in the subsequent stage.

We design a CNN termed as HEG-S for depth-based scene classification. Table 1 illustrates the detailed network architecture. The first 6 trainable layers are 3×3 convolutional layers. The output feature maps are spatially downsampled to 2×2 using an adaptive max pooling layer and reshaped into a feature vector, which is then consumed by 3 additional fully connected layers. A batch normalization and leaky ReLU layer are appended to each intermediate trainable layer. The final fully connected layer generates a score for each scene class, which is further normalized into a probability via a softmax layer. Given the inferred scene class probabilities and the ground truth labels, HEG-S is trained by optimizing a cross-entropy loss. After training, the output feature maps of the intermediate convolutional layers can be adopted as hierarchical embeddings to compute supervisions for training the depth estimation network.

HEG-R trained on depth reconstruction Depth reconstruction aims to extract representative features from the input depth and restore the depth information from the extracted features. For one thing, it can be learned without additional labels. For another, it is highly relevant to depth estimation since both the input and the target output are depth maps. As a result, we propose to explore depth reconstruction as the second task for training HEGs.

We design a new HEG network with an encoder-decoder architecture for depth reconstruction. The encoder network consists of 6 convolutional layers and 1 fully connected layer. For a fair comparison, the detailed architecture of the encoder (as shown in Table 2) mostly follows that of HEG-S, except that batch normalization after each convolutional layer is discarded due to the reconstruction purpose. The decoder architecture is symmetric to that of the encoder, where only 2×2 strided convolutional layers are replaced by transpose

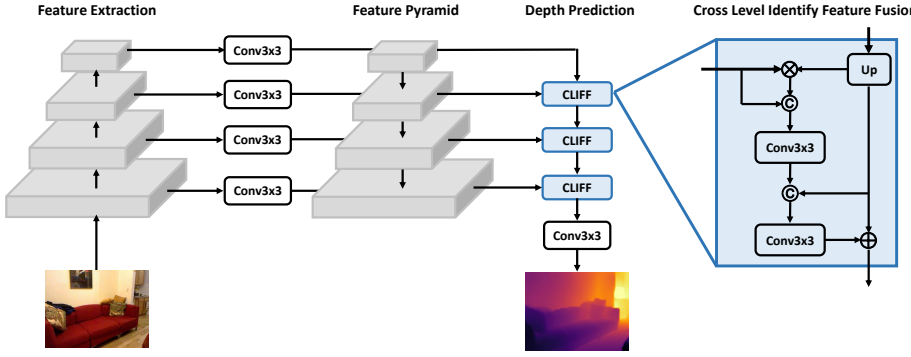

Fig. 1. Overview of the proposed CLIFFNet.

convolutional layers with a $\times 2$ upsampling factor. One of the key ingredients in the proposed network is the 256 dimensional feature vector generated by the encoder, which serves as a bottleneck connecting the encoder and decoder. As the bottleneck structure significantly squeezes the feature dimension, it forces the convolutional layers of the encoder to capture the most representative features from the input depth map, preventing the reconstruction network degenerating into a trivial identity mapping.

We name the above HEG trained on depth reconstruction as HEG-R. The multi-level convolutional feature maps generated by the encoder of HEG-R are investigated as an embedding hierarchy for training depth estimation.

Discussion The proposed hierarchical loss is reminiscent of the perceptual losses which are mainly adopted by generative models to produce photo-realistic results. It has been shown that the perceptual losses can effectively improve the visual quality but may hinder the quantitative performance [12]. In comparison, we focus on analyzing different training tasks and HEG architectures to compute hierarchical depth losses. Our experiment shows that the proposed hierarchical loss can not only benefit the perceptual quality but also significantly improves the quantitative performance in terms of depth metrics. It should also be noted that although our current training strategies are selected according to the proposed two standards, they are not directly coupled with our ultimate goal of finding an optimal embedding space for a hierarchical loss. In our future work, we will explore meta-learning techniques to learn optimal hierarchical embedding spaces for depth supervision.

3.2 CLIFFNet for Depth Estimation

Following most existing works [14, 33], the proposed CLIFFNet performs depth estimation with a fully convolutional architecture, which consists of three components: a feature extraction sub-network, a feature pyramid sub-network, and a depth prediction sub-network. The feature extraction sub-network takes a single

RGB image as input and extracts a collection of multi-level convolutional feature maps of various resolutions. The generated feature maps are then fed into the feature pyramid sub-network through lateral connections, which propagates the semantic information from high-level to low-level feature maps, producing a feature pyramid. The depth prediction sub-network make the final prediction based on the feature pyramid. Figure 1 provides an overview of the architecture.

In order to take full advantage of the feature pyramid, some prior methods adopt a direct fusion strategy. They first upsample all feature maps in the pyramid into the same resolution, which are then combined through concatenation and used to estimate the depth map. Although high-level features with rich semantic information are used to benefit robust predictions, they are directly upsampled from very low-resolutions, leading to blurry depth prediction. An alternative idea is the progressive fusion strategy, where high-level features are gradually upsampled (*e.g.*, by $\times 2$ each time) and combined with lower level features of the same resolution. Though the blurry prediction issue can be alleviated, the output features are dominated by low-level cues which are not robust to challenging scenarios. To address this issue, we propose the cross level identity feature fusion (CLIFF) module, which not only enhances the visual quality but also preserves high-level features to facilitate more robust depth estimation.

CLIFF Module The CLIFF module takes a high-level and low-level feature map as input. We first upsample the high-level feature map using bilinear interpolation to ensure that two input feature maps have the same spatial resolution. Since high-level feature is more reliable with less noise, we refine the low-level feature through an attention mechanism by multiplying it with the high-level feature. As such, accurate responses in the low-level feature are further strengthened, while noisy responses are weakened. In order to achieve the optimal combination of the high-level feature, original and refined low-level feature, these features are further selected through two convolutional layers. Specifically, the first convolutional layer learns to select and aggregate low-level features by taking the concatenation of the original and refined low-level feature as input. Its output is then concatenated with the high-level feature and serves as the input to the second convolutional layer, further allowing feature selection between low-level and high-level feature maps. Finally, to facilitate gradients back-propagation and to preserve high-level semantic cues, an identity mapping from the high-level feature to the output feature is added. Denoting the low-level feature as $\boldsymbol{F}^l$, the upsampled high-level feature as $\boldsymbol{F}^h$, and the output as $\boldsymbol{F}^o$, the above operations can be formally described as:

$$\begin{cases} \boldsymbol{F}^o = \boldsymbol{F}_2^c + \boldsymbol{F}^h, \\ \boldsymbol{F}_2^c = \boldsymbol{W}_2 * [\boldsymbol{F}_1^c, \boldsymbol{F}^h] + \boldsymbol{b}_2, \\ \boldsymbol{F}_1^c = \boldsymbol{W}_1 * [\boldsymbol{F}^l, \boldsymbol{F}^a] + \boldsymbol{b}_1, \\ \boldsymbol{F}^a = \boldsymbol{F}^l \odot \boldsymbol{F}^h, \end{cases} \tag{2}$$

where $\boldsymbol{F}_i^c$ denotes the selected feature using convolutional layers parameterized by weight $\boldsymbol{W}_i$ and bias $\boldsymbol{b}_i$. $[\cdot, \cdot]$ indicates the concatenation of two feature maps

along the channel dimension. The operators $*$ and $\odot$ indicate convolution and element-wise multiplication, respectively.

In the proposed depth prediction sub-network, the CLIFF modules are repeatedly applied to gradually perform feature fusion from high-level to low-level features. The fused feature generated by the last CLIFF module is then fed into a convolutional layer to produce the final depth prediction.

4 Experiments

4.1 Implementation Details

We compute L1 losses on embedding spaces for training depth estimation and exhaustively search the optimal combination of embedding spaces generated by the proposed HEGs. Our empirical results (See supplementary materials for details) show that the combination of the original depth space and the embedding spaces generated by the 2nd and 4th layer of HEGs delivers the best performance when used for hierarchical loss computation. The result is consistent to both HEG-S and HEG-R, giving rise to our final loss function as below:

$$L_D(\boldsymbol{d}, \hat{\boldsymbol{d}}) = \sum_{k=\in\{0,2,4\}} w_k \|G_k(\boldsymbol{d}) - G_k(\hat{\boldsymbol{d}}\|_1, \tag{3}$$

where the loss weights are determined through grid-search and fixed as $w_0 = 1.0, w_2 = 10.0, w_4 = 15.0$.

For the proposed CLIFFNet, we adopt the first 5 residual block of a pre-trained ResNet-50 network [9] as the feature extraction sub-network. The feature pyramid sub-network are designed closely following [19] (See supplementary materials for architecture details). We resize each input image to have a minimum side of 228 pixels by maintaining its aspect ratio. All the networks are trained using Adam optimizer [13] with a batch size of 8 images and initial learning rates 1e-3, 1e-4, and 1e-4 for HEG-S, HEG-R, and CLIFFNet, respectively. Source code will be made publicly available[6].

Our experiments are conducted on NYU-Depth V2 [27] and Cityscapes [3] dataset. The NYU-Depth V2 dataset contains 464 indoor scenes, where 249 of them are for training and the rest for testing. 40K image-depth pairs are sampled from all the 120K training samples. We first use the sampled depth to train HEG-S for 249 scene classification and HEG-R for depth reconstruction, then use the trained HEGs to compute losses to learn CLIFFNet for depth estimation. On one NVIDIA 1080Ti GPU, The training processes of HEG-S and HEG-R take around 4 hours, respecitvely, while the depth network is trained for around 22 hours. The depth network with 36.89M parameters runs at 37.86 FPS during inference. As in [6, 7], we evaluate the proposed method using 7 widely adopted metrics defined in Table 3. We compute these metrics using the implementation provided by [7] Due to page limits, we present the results on Cityscapes in the supplementary materials.

[6] https://github.com/scott89/CLIFFNet

Table 3. Adopted evaluation metrics for depth estimation. d_i and $\hat{d}_i$ denote the ground truth and estimated depth value of pixel i. N denotes the total number of pixels.

Metric	Definition	Metric	Definition
RMSE	$(\frac{1}{N}\sum_i(\hat{d}_i - d_i)^2)^{(\frac{1}{2})}$	RMSE (log)	$(\frac{1}{N}\sum_i(\log\hat{d}_i - \log d_i)^2)^{(\frac{1}{2})}$
Abs Rel	$\frac{1}{N}\sum_i \lvert\hat{d}_i - d_i\rvert/d_i$	Sq Rel	$\frac{1}{N}\sum_i(\hat{d}_i - d_i)^2/d_i$
Pn	Percentage of d_i such that $\max\{\frac{d_i}{\hat{d}_i}, \frac{\hat{d}_i}{d_i}\} < 1.25^n$		

Table 4. Comparison with state-of-the-art methods on NYU-Depth V2 dataset [27]. The best and second best results are in **bold** font and underlined, respectively.

Method	Error				Accuracy		
	RMSE	RMSE (log)	Abs Rel	Sq Rel	P1	P2	P3
Eigen *et al.* [6]	0.874	0.284	0.218	0.207	0.616	0.889	0.971
Liu *et al.* [21]	0.756	0.261	0.209	0.180	0.662	0.913	0.979
Eigen and Fergus [5]	0.874	0.284	0.218	0.207	0.616	0.889	0.971
Laina *et al.* [14]	0.584	0.198	0.136	0.101	0.822	0.956	0.989
Chakrabarti *et al.* [1]	0.620	0.205	0.149	0.118	0.806	0.958	0.987
Xu *et al.* [34]	0.593	-	0.125	–	0.806	0.952	0.986
Qi *et al.* [24]	0.569	-	0.128	–	0.834	0.960	0.990
Lee *et al.* [15]	0.572	0.193	0.139	0.096	0.815	0.963	**0.991**
Fu *et al.* [7]	0.509	0.188	**0.116**	**0.089**	0.828	**0.965**	0.986
Xu *et al.* [33]	0.582	-	0.120	–	0.817	0.954	0.987
CLIFFNet-R	0.497	0.180	0.129	**0.089**	0.841	0.963	**0.991**
CLIFFNet-S	**0.493**	**0.171**	0.128	**0.089**	**0.844**	0.964	**0.991**

4.2 Comparison to State of the Arts

On NYU-Depth V2, we compare with 12 state-of-the-art methods. The quantitative results are reported in Table 4, where CLIFFNet-R and CLIFFNet-S denote the proposed CLIFFNet trained with hierarchical losses computed using HEG-R and HEG-S, respectively. Both CLIFFNet-R and CLIFFNet-S compare favorably against state-of-the-art methods. Among others, CLIFFNet-S consistently outperforms the other methods and achieves the top performance in terms of 5 metrics. Though the performance of CLIFFNet-R is slightly worse than CLIFFNet-S, it also delivers state-of-the-art performance in terms of all the 7 metrics. In particular, its performance in terms of RMSE, RMSE (log) and P2 is comparable to CLIFFNet-R. Among the other compared methods, Lee *et al.* [16] and Fu *et al.* [7] also achieve outstanding performances. However, it should be noted that both Lee *et al.* [16] and Fu *et al.* [7] use all the 120K training images, while our proposed method and most other approaches [33, 35, 34] use only a subset of the training data.

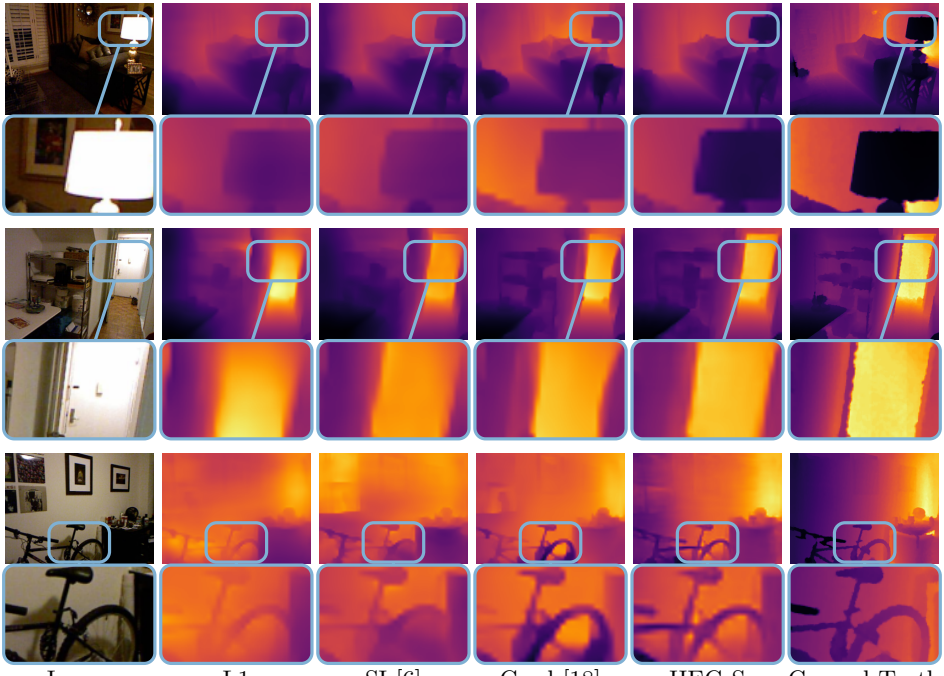

Fig. 2. Depth maps predicted using different loss functions.

Table 5. Comparison of different losses on NYU-Depth V2 dataset [27]. The best results are in **bold** font.

Methods	L1	Grad	SI	DA	MT	HEG-Rn	HEG-Im	HEG-R	HEG-S
RMS	0.529	0.513	0.520	0.511	0.530	0.517	0.523	0.497	**0.493**
Abs Rel	0.135	0.132	0.134	0.130	0.134	0.134	0.132	0.129	**0.128**
P1	0.817	0.830	0.820	0.835	0.815	0.815	0.829	0.841	**0.844**
P2	0.961	0.964	0.963	0.964	0.960	0.961	0.963	**0.963**	**0.964**

4.3 Ablation Study

Effectiveness of Hierarchical Loss To further verify the effectiveness of the proposed hierarchical losses, we evaluate the performance of our CLIFFNet variants trained with different losses. Since the network architectures are the same, we refer to different variants using the name of the adopted loss function. Among them, L1 represents only using the L1 loss computed on the original depth space, while all the other variants combine the depth space L1 loss with other form of loss functions. Specifically, Grad indicates the combination of depth space L1 loss and depth gradient loss [18]. SI denotes the scale invariant loss [6]. DA indicates the depth aware loss. HEG-S and HEG-R represent the hierarchical losses proposed in Section 3.1. HEG-Rn denotes the proposed hierarchical loss com-

puted using a randomly initialized HEG. HEG-Im indicates the hierarchical loss computed by a HEG with the same architecture as HEG-S trained on ImageNet classification task [4]. The input channels of the kernels on the first convolutional layer are averaged in order to take depth map as input.

Table 5 shows the comparison results of different variants on NYU-Depth V2 dataset. The comparison of L1 against SI and DA confirms the advantage of loss functions defined on additional embeddings over those computed only on the original depth space. However, compared with the hand-designed losses SI and DA, the proposed hierarchical embeddings generated by HEG-S and HEG-R are learned in a data-driven manner, leading to more superior performance. The proposed HEG-S and HEG-R trained on carefully designed tasks significantly outperform the randomly initialized HEG-Rn and HEG-Im trained on irrelevant image classification. Figure 2 shows the predicted depth maps of our CLIFFNet trained with different losses. It can be observed that the predictions using HEG-S are perceptually more realistic than other losses. The performance of HEG-Rn and HEG-Im further justifies the importance of seeking relevant tasks for learning loss embeddings.

One may wonder that the advantages of HEG-S may be caused by using additional scene labels. To verify this, we design another variant model named MT, which adds an additional scene classification module on top of the Res-5 feature map generated by the feature extraction sub-network. It consists of a global average pooling followed by two fully connected layers. We then train MT on both depth estimation (using depth space L1 loss) and scene classification (using cross-entropy loss) in a multi-task training manner. As illustrated in Table 5, the depth estimation performance of MT trained on additional scene classification task is similar to that of the baseline L1, and is worse to our proposed HEG-R and HEG-S by a considerable margin. The results suggest that compared to muti-task learning the proposed HEG-S can serve as a more superior strategy to benefit depth estimation with additional scene classification annotations.

Visualization of Loss Gradients To gain more intuitive understanding of the hierarchical losses, we perform additional visualization experiments to analyze the impact of different loss functions on network training. During one intermediate training iteration, we forward-propagate an input image through CLIFFNet, compute different losses using the predicted and ground truth depth maps, and then back-propagate the gradients of the loss functions to the predicted depth space. Figure 3 provides the visualization of depth space gradients back-propagated from different loss functions. It can be observed that the gradient magnitude of L1 is almost uniform in each pixels, while DA [11] attaches a higher weight on distant regions with larger depth values. The Grad loss focuses more on the low-level boundary regions. In comparison, the gradients of HEG-S demonstrates more clear hierarchical patterns. The behavior of HEG-S2 (with loss computed in the 2nd layer of HEG-S) is similar to Grad, but seems to be more robust to noisy depth edges. Meanwhile, HEG-S4 (with loss computed in the 4th layer of HEG-S) focuses on more on interior of object regions with

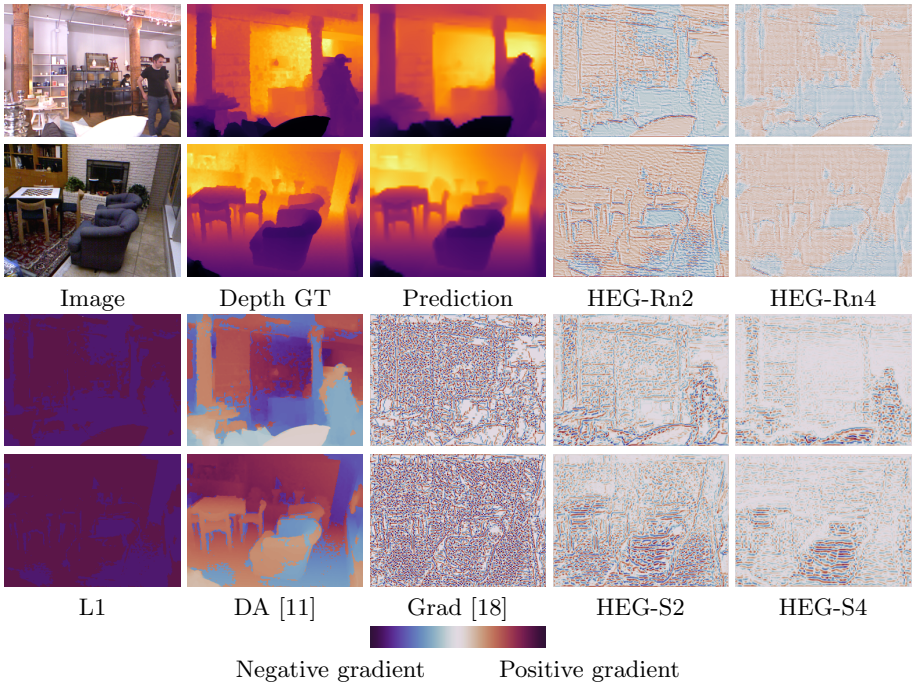

Fig. 3. Gradients backpropagated to the predicted depth space from different losses. L1: L1 loss computed on the original depth space. DA: depth aware loss [11]. Grad: depth gradient loss [18]. HEG-Rn: L1 loss on the embedding produced by the i-th layer of a randomly initialized HEG. HEG-Si: L1 loss on the embedding produced by the i-th layer of a HEG pre-trained on scene classification.

semantic meaning. Compared with HEG-S, the gradients of randomly initialized HEG-Rn fail to exhibit such hierarchical patterns. The above observations on HEG-S also hold for HEG-R. According to their behaviors, we conjecture that the proposed hierarchical losses is able to capture multi-scale contexts and therefore more robust to local noise labels and outliers.

Impact of CLIFF Module The core architecture designs of the proposed CLIFF module include a) attention based low-level feature refinement, b) multi-level feature selection, and c) identity mapping of high-level features. We ablate these core designs by comparing 4 variants of CLIFF module for cross level feature fusion. Among them, CLIFF-w/o-att discards attention based feature refinement. It select the input features by applying two convolutional layers on their concatenation. An identity mapping of high-level features is then added to the selected feature to produce the output. CLIFF-w/o-sel discards feature selection, where two convolutional features are directly applied to the sum of the original and refined low-level features and high-level features. CLIFF-w/o-id removes the identity mapping. Finally, a baseline module that does not contain

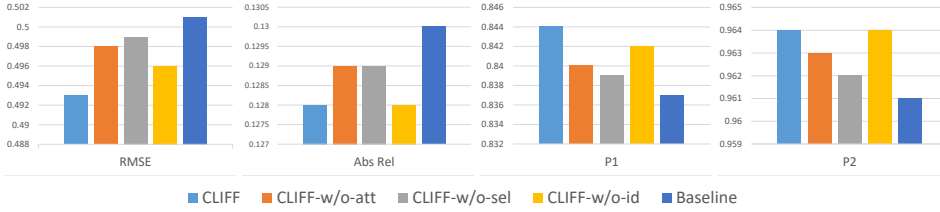

Fig. 4. Comparison results of different CLIFF variants on NYU-Depth V2 in terms of errors (the first row) and accuracy (the second row).

any of the above 3 architecture design is developed. It combines two input feature maps through addition and then produces the output features with two convolutional layers.

We apply the above 4 variants to the depth prediction module in the same way as the proposed CLIFF module, leading to 4 variants of the proposed method. We then train the 4 variants as well as the proposed CLIFFNet using hierarchical embedding losses computed by HEG-S. Figure 4 demonstrates the comparison results on NYU-Depth V2 dataset. It can be observed that each of the three core architecture designs can effectively improve depth estimation performance. By combining all the architecture designs, CLIFF outperforms the baseline for a large margin, suggesting the contribution of each design is relative orthogonal to the others. We also performs additional ablation studies to investigate the performance of intermediate output from CLIFF modules. We leave the detailed results in the supplementary materials due to page limits.

5 Conclusion

We propose hierarchical losses for monocular depth estimation. Rather than defined on a per pixel basis, they are computed in hierarchical embedding spaces and can be automatically learned from training data. To obtain superior hierarchical embeddings, we design two embedding generators, named as HEG-S and HEG-R, which are trained on scene classification and depth reconstruction, respectively. Experiments show that learned hierarchical losses can capture multi-scale contexts and are more robust to outliers, leading to significant performance gain. In addition, we further propose CLIFFNet for depth estimation, which provides a more effective manner for cross level feature fusion. CLIFFNet trained with hierarchical losses sets new record on popular benchmarks.

Acknowledgements This work is supported by National Key R&D Program of China (2018AAA0102001), National Natural Science Foundation of China (61725202, U1903215, 61829102, 91538201, 61771088, 61751212, 61906031), Fundamental Research Funds for the Central Universities (DUT19GJ201), Dalian Innovation Leaders Support Plan (2018RD07), China Postdoctoral Science Foundation (2019M661095), National Postdoctoral Program for Innovative Talent (BX20190055).

References

1. Chakrabarti, A., Shao, J., Shakhnarovich, G.: Depth from a single image by harmonizing overcomplete local network predictions. In: Lee, D.D., Sugiyama, M., von Luxburg, U., Guyon, I., Garnett, R. (eds.) NIPS. pp. 2658–2666 (2016)
2. Chen, W., Fu, Z., Yang, D., Deng, J.: Single-image depth perception in the wild. In: NIPS. pp. 730–738 (2016)
3. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The cityscapes dataset for semantic urban scene understanding. In: CVPR. pp. 3213–3223 (2016)
4. Deng, J., Dong, W., Socher, R., Li, L., Li, K., Li, F.: Imagenet: A large-scale hierarchical image database. In: CVPR. pp. 248–255 (2009)
5. Eigen, D., Fergus, R.: Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In: ICCV. pp. 2650–2658 (2015)
6. Eigen, D., Puhrsch, C., Fergus, R.: Depth map prediction from a single image using a multi-scale deep network. In: NIPS. pp. 2366–2374 (2014)
7. Fu, H., Gong, M., Wang, C., Batmanghelich, K., Tao, D.: Deep ordinal regression network for monocular depth estimation. In: CVPR. pp. 2002–2011 (2018)
8. Godard, C., Aodha, O.M., Firman, M., Brostow, G.J.: Digging into self-supervised monocular depth estimation. In: ICCV. pp. 3827–3837 (2019)
9. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. pp. 770–778 (2016)
10. Hoiem, D., Efros, A.A., Hebert, M.: Geometric context from a single image. In: ICCV. pp. 654–661 (2005)
11. Jiao, J., Cao, Y., Song, Y., Lau, R.: Look deeper into depth: Monocular depth estimation with semantic booster and attention-driven loss. In: ECCV. pp. 53–69 (2018)
12. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV. vol. 9906, pp. 694–711 (2016)
13. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
14. Laina, I., Rupprecht, C., Belagiannis, V., Tombari, F., Navab, N.: Deeper depth prediction with fully convolutional residual networks. In: 3DV. pp. 239–248 (2016)
15. Lee, J.H., Heo, M., Kim, K.R., Kim, C.S.: Single-image depth estimation based on fourier domain analysis. In: CVPR. pp. 330–339 (2018)
16. Lee, J.H., Kim, C.S.: Monocular depth estimation using relative depth maps. In: CVPR. pp. 9729–9738 (2019)
17. Li, J., Klein, R., Yao, A.: A two-streamed network for estimating fine-scaled depth maps from single rgb images. In: ICCV. pp. 3372–3380 (2017)
18. Li, Z., Snavely, N.: Megadepth: Learning single-view depth prediction from internet photos. In: CVPR. pp. 2041–2050 (2018)
19. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: CVPR. pp. 2117–2125 (2017)
20. Liu, B., Gould, S., Koller, D.: Single image depth estimation from predicted semantic labels. In: CVPR. pp. 1253–1260 (2010)
21. Liu, F., Shen, C., Lin, G.: Deep convolutional neural fields for depth estimation from a single image. In: CVPR. pp. 5162–5170 (2015)
22. Liu, M., Salzmann, M., He, X.: Discrete-continuous depth estimation from a single image. In: CVPR. pp. 716–723 (2014)

23. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: CVPR. pp. 3431–3440 (2015)
24. Qi, X., Liao, R., Liu, Z., Urtasun, R., Jia, J.: Geonet: Geometric neural network for joint depth and surface normal estimation. In: CVPR. pp. 283–291 (2018)
25. Rad, M.S., Bozorgtabar, B., Marti, U.V., Basler, M., Ekenel, H.K., Thiran, J.P.: Srobb: Targeted perceptual loss for single image super-resolution. In: ICCV. pp. 2710–2719 (2019)
26. Saxena, A., Sun, M., Ng, A.Y.: Make3d: Learning 3d scene structure from a single still image. TPAMI **31**(5), 824–840 (2008)
27. Silberman, N., Hoiem, D., Kohli, P., Fergus, R.: Indoor segmentation and support inference from rgbd images. In: ECCV. pp. 746–760 (2012)
28. Wang, L., Ouyang, W., Wang, X., Lu, H.: Visual tracking with fully convolutional networks. In: ICCV. pp. 3119–3127 (2015)
29. Wang, L., Shen, X., Zhang, J., Wang, O., Lin, Z.L., Hsieh, C., Kong, S., Lu, H.: Deeplens: shallow depth of field from a single image. ACM Trans. Graph. **37**(6), 245:1–245:11 (2018)
30. Wang, L., Zhang, J., Wang, O., Lin, Z., Lu, H.: SDC-Depth: Semantic divide-and-conquer network for monocular depth estimation. In: CVPR (June 2020)
31. Watson, J., Firman, M., Brostow, G.J., Turmukhambetov, D.: Self-supervised monocular depth hints. In: ICCV. pp. 2162–2171 (2019)
32. Xian, K., Shen, C., Cao, Z., Lu, H., Xiao, Y., Li, R., Luo, Z.: Monocular relative depth perception with web stereo data supervision. In: CVPR. pp. 311–320 (2018)
33. Xu, D., Ouyang, W., Wang, X., Sebe, N.: Pad-net: Multi-tasks guided prediction-and-distillation network for simultaneous depth estimation and scene parsing. In: CVPR. pp. 675–684 (2018)
34. Xu, D., Wang, W., Tang, H., Liu, H., Sebe, N., Ricci, E.: Structured attention guided convolutional neural fields for monocular depth estimation. In: CVPR. pp. 3917–3925 (2018)
35. Zhang, Z., Cui, Z., Xu, C., Jie, Z., Li, X., Yang, J.: Joint task-recursive learning for semantic segmentation and depth estimation. In: ECCV. pp. 235–251 (2018)
36. Zhi, S., Bloesch, M., Leutenegger, S., Davison, A.J.: Scenecode: Monocular dense semantic reconstruction using learned encoded scene representations. In: CVPR. pp. 11776–11785 (2019)