Attention Attention Everywhere: Monocular Depth Prediction with Skip Attention

Ashutosh Agarwal Chetan Arora
Indian Institute of Technology Delhi

Abstract

Monocular Depth Estimation (MDE) aims to predict pixel-wise depth given a single RGB image. For both, the convolutional as well as the recent attention-based models, encoder-decoder-based architectures have been found to be useful due to the simultaneous requirement of global context and pixel-level resolution. Typically, a skip connection module is used to fuse the encoder and decoder features, which comprises of feature map concatenation followed by a convolution operation. Inspired by the demonstrated benefits of attention in a multitude of computer vision problems, we propose an attention-based fusion of encoder and decoder features. We pose MDE as a pixel query refinement problem, where coarsest-level encoder features are used to initialize pixel-level queries, which are then refined to higher resolutions by the proposed Skip Attention Module (SAM). We formulate the prediction problem as ordinal regression over the bin centers that discretize the continuous depth range and introduce a Bin Center Predictor (BCP) module that predicts bins at the coarsest level using pixel queries. Apart from the benefit of image adaptive depth binning, the proposed design helps learn improved depth embedding in initial pixel queries via direct supervision from the ground truth. Extensive experiments on the two canonical datasets, NYUV2 and KITTI, show that our architecture outperforms the state-of-the-art by 5.3% and 3.9%, respectively, along with an improved generalization performance by 9.4% on the SUNRGBD dataset. Code is available at https://github.com/ashutosh1807/PixelFormer.git.

1. Introduction

Monocular Depth Estimation (MDE) is a well-studied topic in computer vision. State-of-the-art (SOTA) techniques for MDE are based on encoder-decoder style Convolutional Neural Network (CNN) architectures [4, 5, 17, 18, 21, 34]. Due to the inherently local nature of a convolution kernel, early-stage feature maps have higher resolution but lack a global receptive field. The feature pyramidal-based decoder mitigates the issue by fusing low-resolution, semantically rich decoder features with the higher resolution but semantically weaker encoder features via a top-down path-way and lateral connections called skip connections [19]. Inline with the recent success of transformers, many latest works have used a self-attention based architectures for MDE [1, 2, 33, 35]. Self-attention increases the receptive field and allows to capture long-range dependencies in feature maps. Practically, it is challenging to use self-attention for high-resolution feature maps due to memory and computational constraints. Hence, the current SOTA [35] uses window based attention using Swin transformer-based encoder backbone [20] to improve efficiency.

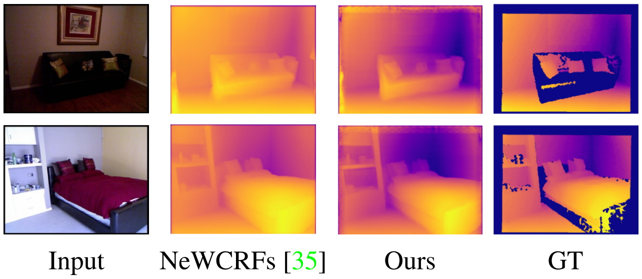

Figure 1: We observe that depth boundaries in state-of-the-art [35] align well with object boundaries, but the depth label is often incorrect. Note the confusion for the middle pillow in the first row and the bed in the second row. We propose skip attention module for fusing long range context from the decoder into the encoder features, which successfully mitigates the discrepancy.

We observe that SOTA [2, 35] techniques are highly accurate in aligning depth edges with the object boundaries. However, there exists a confusion in giving the depth label to a pixel (c.f. Fig. 1). We posit this due to the inability of current techniques to effectively fuse high-resolution local features from the encoder and global contextual features from the decoder. Typically such a fusion is achieved through a skip connection module implementing feature concatenation followed by a convolution operation. Weights of the convolution kernels are highly localized, which restricts the flow of semantic information from long ranges affecting the ability of the model to predict the correct depth label for a pixel. To mitigate the constraint, we introduce a skip-attention module (SAM) that helps integrate information using window-based cross-attention. SAM calculates self-similarity between pixel queries based on decoder features and their corresponding neighbors from the encoder features in a predefined window to attend to and aggregate information at a longer range. We implement the overall architecture as a pixel query refinement problem. We use the coarsest feature map from the encoder with maximum global information to initialize pixel queries using a Pixel Query Initialiser Module. The pixel queries are then refined with the help of a SAM module to finer scales.

Recent MDE techniques [2] formulate the problem as a classification-regression one, in which the depth is predicted by a linear combination of bin centers discretized over the depth range. The bin centers are predicted adaptively per image, allowing the network to concentrate on the depth range regions that are more likely to occur in the scene of the input image. A vision transformer that aggregates global information from the output of another encoder-decoder-based transformer model is typically used to generate the bin centers. Since we pose MDE as a pixel query refinement problem starting from the coarsest resolution, we propose a lightweight Bin Center Module (BCP) that predicts bin centers based on the initial pixel queries. This is more efficient than decoding features and then attending again in current SOTA [2]. The proposed design also helps embed the depth information into the initial pixel queries via direct ground truth supervision.

Contributions: The specific contributions of this work are as follows: **(1)** We propose a novel strategy for predicting depth using a single image by viewing it as a pixel query refinement problem. **(2)** We introduce a Skip Attention Module (SAM) that uses a window-based cross-attention module to refine pixel queries from the decoder feature maps for cross-attending to higher resolution encoder features. **(3)** We present a Bin Center Predictor (BCP) Module that estimates bin centers adaptively per image using the global information from the coarsest-level feature maps. This helps to provide direct supervision to initial pixel queries from ground truth depth, leading to better query embedding. **(4)** We combine the novel design elements in an encoder-decoder framework comprised of a vision transformer backbone. The proposed architecture called PixelFormer achieves state-of-the-art (SOTA) performance on indoor NYUV2 and outdoor KITTI datasets, improving the current SOTA by 5.3% and 3.9%, in terms of absolute relative error and square relative error, respectively. Additionally, PixelFormer improves the generalization performance by 9.4% over SOTA on the SUNRGBD dataset in terms of absolute relative error.

2. Related Works

CNN based MDE Techniques: Eigen *et al*. [4] first utilized CNN to predict depth from a single image by integrating global and local information. Song *et al*. [29] have proposed a Laplacian pyramid-based model, and CLIFFNet [31] a multi-scale convolutional fusion architecture to generate high-quality depth prediction. Yin *et al*. [34] introduced a geometric constraint named *virtual normal*, and Naderi *et al*. [21] proposed similarity between the RGB image and the corresponding depth map at the geometric edges to regularize predicted depth. Lee *et al*. [18] enforce a model to learn structural information about the scene by learning the relationship between image patches close to each other. Whereas, Patil *et al*. [24] exploit coplanar pixels to improve the predicted depth.

Transformer based MDE Models: Recent works have used Vision Transformer (ViT) architecture to improve the receptive field of a CNN in lower layers. Ranftl *et al*. [25] uses a CNN to extract feature maps at $\left(\frac{1}{16}\right)^{\text{th}}$ resolution, which are passed to a vision transformer for global information aggregation. Bhat *et al*. [2] uses a CNN-based encoder-decoder backbone and a ViT model to predict adaptive bins and pixel-level depths. NeWCRFs [35] uses Swin Transformer backbone [20] with CRFs at multiple scales.

MDE as Classification Vs Regression Task: Modeling MDE as a regression problem leads to suboptimal solutions and faces convergence issues. Huan *et al*. [5] first introduced the depth prediction task as a classification-regression problem solved by a CNN-based classification network in which the depth is predicted as a linear combination of bin centers discretized over the depth range. Recently, Bhat *et al*. [2] proposed the prediction of bin centers adaptively per image using a ViT transformer on top of a transformer-based encoder-decoder backbone. In this work, we propose a single encoder-decoder backbone with a lightweight BCP module to predict bin centers using the coarsest resolution encoder feature maps.

Skip Connections: Skip connections were introduced by UNet [26] to forward high-resolution information from the encoder to the decoder via feature fusion. However, a naive fusion of early encoder and late decoder information is hindered by their semantic gap [36]. MultiResUnet [13] replaces simplistic skip connection with a series of residual blocks to alleviate the semantic gap. Attention U-Net [22] suppresses irrelevant regions in an input image while highlighting salient features useful for a specific task before the feature fusion. SANet [32] uses attention to complete the point cloud at the decoder stage by injecting information using the encoder. In this work, we use skip attention to retrieve high-resolution details from encoder features using global contextual queries based on the decoder features.

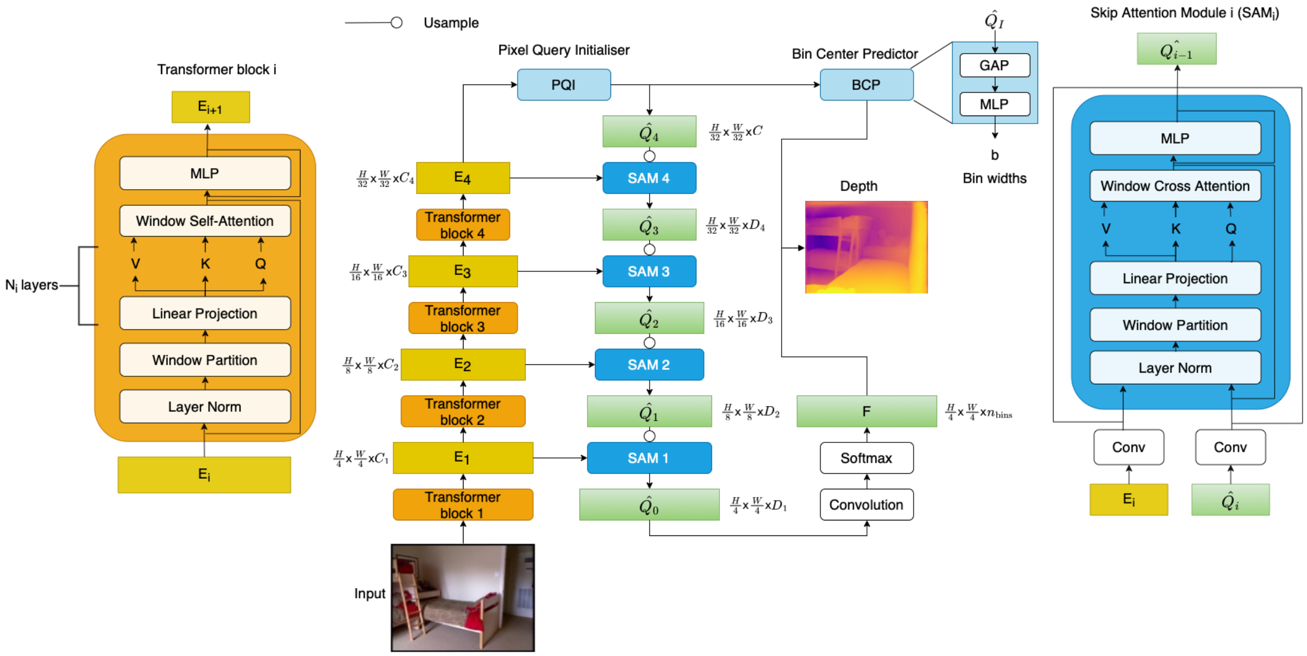

Figure 2: **Detailed Architecture of our proposed approach PixelFormer :** Given an input image, a vision transformer-based encoder first extracts the multiscale feature maps. The feature map with the coarsest resolution (E_4) is given as input to the PQI module. The PQI module produces initial pixel queries that are given as input to the BCP module that produces the bin widths. The initial pixel queries are then refined to higher resolutions using the SAM modules. Finally, a convolution operation followed by a *softmax* is applied to get probability distribution per pixel over the bin centers.

3. Proposed Methodology

Problem Definition: Following [2, 5], we model MDE as a classification-regression task. Given an input image I, the network predicts bin widths, b, that discretize continuous depth range into an n_{bins} number of intervals. The bins are predicted adaptively for each image. The final n_{bins} dimensional probability vector is treated as the weight vector, and the depth, d_i, at a pixel i, is computed as a linear combination of the probability scores at the pixel with the predicted per-image bin-centers.

Architecture Overview: The input image I is first fed to a Swin Transformer [20] that uses multiple layers of window-based self-attention to extract the feature maps representing the image at resolution scale $\{\frac{1}{4}, \frac{1}{8}, \frac{1}{16}, \frac{1}{32}\}$ w.r.t. I. The feature maps have a global receptive field due to the inherent nature of a ViT backbone. The feature map at $\left(\frac{1}{32}\right)^{\text{th}}$ scale is then fed to the proposed Pixel Query Initialiser (PQI) module. The PQI module aggregates the entire scene information using multi-scale global average pooling to initialize *pixel queries*. The pixel queries are hierarchically refined with the encoder feature maps using the proposed *Skip Attention Module* (SAM) deployed at various stages to predict per-pixel probability distribution over the bin centers. The initialized pixel queries are also sent to the *Bin Center Predictor* (BCP) proposed in this work. BCP predicts bin centers adaptively per image using global average pooling followed by MLP layers. Fig. 2 gives a pictorial description.

Pixel Query Initialiser (PQI): The Pixel Query Initialiser (PQI) module aggregates the global information of the scene into each pixel-level embedding. The image feature map with the coarsest resolution, which contains the most essential details in the scene, is fed as an input to the PQI module. Given an input feature map of size $\frac{H}{32} \times \frac{W}{32} \times C_4$, the PQI module uses pyramid spatial pooling (PSP) [11] with an adaptive global pooling at scales 1, 2, 3, and 6. The feature maps are then upsampled to $\frac{1}{32}^{th}$ scale and concatenated. A convolution operation is then performed to integrate the global information effectively, as in [35], to get initial pixel queries Q_I of size $\frac{H}{32} \times \frac{W}{32} \times C$, where C = 512.

Bin Center Predictor (BCP): Previous works [2] have used a vision transformer (ViT) to predict bin centers that discretize the image depth into a fixed number of intervals. ViT divides the image feature map into 16 × 16 patches and uses self-attention layers to exchange information among the patches. The first embedding is passed through an MLP head to predict the bin centers. Instead of decoding the feature map to high resolution and then using ViT, we propose to use the initial pixel queries to predict the bin centers. Apart from being more efficient, the proposed design helps in embedding the depth information into the pixel queries via direct ground truth supervision. Our BCP module consists of a simple Global Average Pooling followed by an MLP layer to predict the bin widths b of dimension n_{bins}. Here, n_{bins} denotes the number of adaptive bins per image. We use $n_{\text{bins}} = 256$ for our model as suggested in [2]. Given the pixel queries Q_I of size $\frac{H}{32} \times \frac{W}{32} \times C$, we predict:

$$b = \texttt{MLP}(\texttt{GAP}(Q)) \tag{1}$$

Finally, the bin centers for the input image are computed as:

$$c\left(b_i\right) = d_{\text{min}} + \left(d_{\text{max}} - d_{\text{min}}\right)\left(\frac{b_i}{2} + \sum_{j=1}^{i-1} b_j\right), \tag{2}$$
$$i \in \{1, \ldots, n_{\text{bins}}\}$$

Skip Attention Module (SAM) Overview: For a dense estimation task, coarse-level semantic features and fine-level details are both critical for an accurate estimation. Hence, similar to previous works [19, 35], we also use a bottom-down approach that starts with the lowest resolution feature map, upsample it, and injects the fine-level details from the encoder feature map at a particular scale using skip connections.

Typically skip connections use convolution operation after concatenating the encoder-decoder features. Unlike convolution operation to fuse the encoder-decoder features where the kernel weights are not adaptive as per the pixel location, we use the skip attention module (SAM), which uses self-similarity between the pixel queries and the corresponding encoder feature map, to effectively fuse the global-local features.

SAM Implementation: Given a pixel query map $\hat{Q}_i$ and the corresponding encoder features E_i for a particular scale i, we first perform a 3×3 convolution E_i with D_i channels on both E_i and $\hat{Q}_i$ so that the number of channels of the pixel queries generated from the decoder features is the same as the number of channels in the encoder feature maps. Post the convolutional operation, a query matrix Q is obtained from $\hat{Q}_i$, and the key K and value V matrices are obtained from E_i using the weights W_q, W_k and W_v implemented using MLP layers. Since it's not computationally feasible for a query q_i corresponding to the pixel query at location i to attend to all the keys in the matrix K, we restrict the attention to a window as suggested for Swin Transformer [20]. Q, K, and V matrices are first divided into windows of size $W \times W$. Similar to [20], we use $W = 7$. Let Q_w, K_w, and V_w be the query, key, and the value corresponding to the pixels in window w.

We compute the output as follows:

$$\text{Attention}(Q, K, V) = \text{Rearrange}(\text{Softmax}(Q_w K_w^T + B) V_w).$$

Here, B denotes relative position bias. B is a learnable matrix of size $w^2 \times w^2$, representing the relative position embedding corresponding to each query and key pair. The attention is computed for each window w after which the rearrange operation places the windows as per their respective spatial location in Q.

To embed the information corresponding to the various depth ranges, each pixel query is divided into H_i heads, and the attention operation is applied for each head. Post attention, per pixel depth embeddings are aggregated using MLP layers. The residual connections post attention and MLP layers are added for smooth gradient flows. To summarise, given $\hat{Q}_i$ and E_i for pixel query and encoder at level i:

$$\begin{aligned}
\bar{Q}_i &= \text{LayerNorm}(\hat{Q}_i) \\
\bar{E}_i &= \text{LayerNorm}(E_i) \\
Q &= W_Q \bar{Q}_i, K = W_K \bar{E}_i, V = W_V \bar{E}_i \\
\hat{Q}_{i-1} &= \text{MultiheadAttention}(Q, K, V) + \hat{Q}_i \\
\hat{Q}_{i-1} &= \text{MLP}(\hat{Q}_{i-1}) + \hat{Q}_{i-1} \\
\hat{Q}_{i-1} &= \text{MLP}(\hat{Q}_{i-1}) + \hat{Q}_i + E_i
\end{aligned}$$

We have used $D_1, D_2, D_3, D_4 = \{128,\ 256,\ 512,\ 1024\}$ where D_i corresponds to the number of channels in the convolutional kernel that is applied before the attention-based fusion of encoder and decoder features at stage i. The number of heads $H_1, H_2, H_3, H_4 = \{4,\ 8,\ 16,\ 32\}$ where H_i represents the number of attention heads used in the SAM module at level i. More details can be seen in Fig. 2.

Decoder Architecture: As shown in Fig. 2, we start with an initial pixel query $\hat{Q}_I$ outputted from the PQI module. $\hat{Q}_I$ is upsampled to twice the resolution size using Pixel Shuffle [27] and sent as input to the SAM module along with the corresponding encoder feature E_4. The initial pixel queries are refined to finer resolutions by attending to multiscale encoder feature maps at various resolutions through our proposed SAM module.

For the given pixel query at level $\hat{Q}_i$ and the corresponding encoder feature E_i,

$$\hat{Q}_i = \text{SAM}(\text{Upsample}(\hat{Q}_{i+1}, E_{i+1}) \quad i \in \{0, 1, 2, 3\}.$$

Here, $\hat{Q}_4$ is same as $\hat{Q}_I$. A convolution operation is performed on $\hat{Q}_0$ to produce the final depth embedding F of size $\frac{H}{32} \times \frac{W}{32} \times n_{\text{bins}}$. Finally, a pixel-wise *softmax* operation is applied to obtain the per bin probability distribution p_{bins}:

$$p_{\text{bins}} = \text{Softmax}(\text{Conv}(\hat{Q}_0)) \tag{3}$$

The final depth is predicted by the linear combination of the bin centers weighted by the probability values:

$$d_i = \sum_{k=1}^{n_{\text{bins}}} c\left(b_k\right) p_{ik}, \tag{4}$$

where d_i is the predicted depth at pixel i, $c\left(b_k\right)$ is the k^{th} bin center, n_{bins} are the number of bins, and p_{ik} is the probability for bin center k for a pixel i.

Training loss: Following previous works [2, 35], we use a scaled version of scaled version of the Scale-Invariant loss (SILog) [4] to supervise our network.

Given, the ground truth depth (d_i^*) and the predicted depth (d_i) at a pixel location i, first the logarithmic distance between d_i and d_i^* is calculated as: $g_i = \log(\hat{d}_i) - \log(d_i^*)$. The SIlog loss is then calculated as follows:

$$\mathcal{L}_{\text{SILog}} = \alpha \sqrt{\frac{1}{n}\sum_i g_i^2 - \frac{\lambda}{n^2}\left(\sum_i g_i\right)^2}. \tag{5}$$

Here, n denotes the number of pixels in an image that have the ground truth values available. Following [2], we use λ = 0.85 and α = 10 for all our experiments.

4. Datasets and Evaluation

NYU Depth V2: NYUV2 [28] is an indoor dataset containing 120K RGB and depth pairs with a size of 480×640 acquired as video sequences from 464 indoor scenes using a Microsoft Kinect. We follow the official training/testing split to evaluate our method, where 249 scenes comprising 50K images are used for training, and 654 images from 215 scenes are used for testing. We use the center cropping proposed by Eigen *et al.* [4], with the depth maps having an upper bound of 10 meters. Our network outputs depth prediction with a resolution of 120×160, which we upsample by $4\times$ to match the ground truth resolution during training and testing.

KITTI Dataset: KITTI [8] is an outdoor dataset that consists of stereo images and 3D scans from 61 scenes captured by multiple sensors mounted on top of a moving vehicle. The dataset contains an input RGB image with a 1241×375 pixels resolution, and the LIDAR scans correspond to it. We use the training/testing split defined by [4] that consists of a subset of 26K left view images from the official Kitti dataset for the training and 697 test set images. To evaluate the test set, we use the crop defined by Garg *et al.* [7] with the depth maps having an upper bound of 80 meters. We use bilinear interpolation to upsample the prediction to match the ground truth image resolution.

SUNRGB-D: SUNRGB-D [30] is an indoor dataset collected using various sensors. It comprises 10335 real RGB-D images of room scenes. The training and testing sets contain 5285 and 5050 images, respectively. We use the official test set for evaluation purposes with an upper bound on the depth of 8 meters.

Evaluation Metrics: We use the standard metrics Average relative error (Abs Rel), Root mean squared error (RMSE), Average Log error ($\log_{10}$), Threshold Accuracy (δ_i) at thresholds $\tau_i's = 1.25, 1.25^2, 1.25^3$ used in earlier works [2, 4, 35] to compare our method against state-of-the-art. For KITTI evaluation, we additionally use Square relative error (Sq Rel).

5. Experiments

Implementation Details: The proposed method is implemented in Pytorch [23]. We use Adam optimizer [15] ($\beta's$= 0.9, 0.999), with a batch size of 8 and a weight decay of 10^{-2}. We use 20 epochs for both KITTI and NYUV2 datasets, with an initial learning rate of 4×10^{-5}, which is decreased linearly to 4×10^{-6}. Our model takes 30 minutes per epoch using 4 NVIDIA A100 GPUs. We use various data augmentation techniques like random rotation, horizontal flipping, changing the image brightness, and Cut Depth [14]. We use the pre-trained weights of Swin-L [20] to initialize our encoder backbone. We follow a similar test protocol as in [2, 35], and output final depth values by averaging the predicted depth for the original image and its mirror image.

Results on NYUV2: Table 1 and Fig. 3 show the quantitative and qualitative results, respectively, on the indoor dataset NYUV2 using our approach, named PixelFormer. Following the test protocol of [10, 17], and without additional training data, our method improves the Absolute relative error by 5.3% over SOTA. The performance gain is significant considering the saturated performance of the dataset in recent years. We see an improvement of 9.6% and 3.5% over the recently proposed methods [24] and NeWCRFs, respectively, in terms of the RMSE error. We see in Fig. 3 that PixelFormer produces more accurate depth maps than Adabins and NeWCRFs, which can be attributed to the proposed SAM module, which allows capturing long-range dependencies. In contrast to other approaches, PixelFormer can estimate depth maps corresponding to missing objects, as shown in the third row of Fig. 3.

Results on KITTI: Table 2 and Fig. 4 show the quantitative and qualitative results, respectively, on the outdoor dataset KITTI. We see an overall improvement of 3.9% and 2.3% in terms of Sq. Rel and RMSE respectively against the SOTA NeWCRFs [35] on the KITTI Eigen Split. We also compare our method against the previous SOTA approaches on the official KITTI test set. Currently, we rank 1^{st} on the official

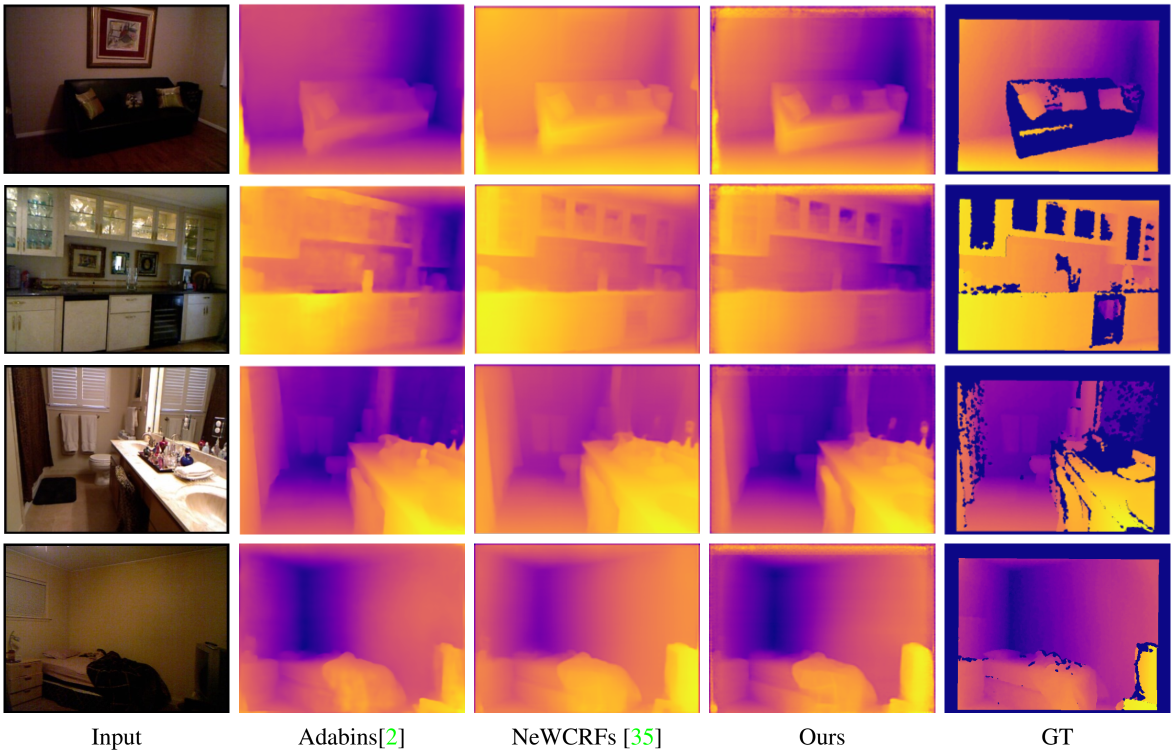

Figure 3: Qualitative comparison of our proposed method PixelFormer on the indoor dataset NYUV2 against Adabins and NeWCRFs.

Method	Venue	Abs Rel↓	RMSE↓	$\log_{10}$↓	δ_1↑	δ_2↑	δ_3↑
Eigen *et al.* [4]	NIPS'14	0.158	0.641	-	0.769	0.950	0.988
DORN [5]	CVPR'18	0.115	0.509	0.051	0.828	0.965	0.992
Yin *et al.* [34]	ICCV'19	0.108	0.416	0.048	0.872	0.976	0.994
BTS [17]	Arxiv'19	0.110	0.392	0.047	0.885	0.978	0.994
DAV [12]	ECCV'20	0.108	0.412	–	0.882	0.980	0.996
TransDepth [33]	ICCV'21	0.106	0.365	0.045	0.900	0.983	0.996
DPT* [25]	ICCV'21	0.110	0.367	0.045	0.904	0.988	**0.998**
PackNet-SAN* [10]	CVPR'21	0.106	0.393	–	0.892	0.979	0.995
Adabins [2]	CVPR'21	0.103	0.364	0.044	0.903	0.984	0.997
Naderi *et al.* [21]	WACV'22	0.097	0.444	0.042	0.897	0.982	0.996
Lee *et al.* [18]	WACV'22	0.107	0.373	0.046	0.893	0.985	0.997
P3Depth [24]	CVPR'22	0.104	0.356	0.043	0.898	0.981	0.996
NeWCRFs [35]	CVPR'22	0.095	0.334	0.041	0.922	**0.992**	**0.998**
PixelFormer (ours)		**0.090**	**0.322**	**0.039**	**0.929**	0.991	**0.998**

Table 1: Results on NYUV2 [28] Dataset. The best results are in **bold** and second best are underlined. "*" means using additional data for training. ↑ means higher the better and ↓ means lower the better. An upper bound of 10 meters on the ground truth depth map is used for evaluation. All the numbers have been taken from the corresponding papers. We see an overall improvement against the SOTA in terms of almost all the metrics used for evaluation.

benchmark[1] against the previous peer-reviewed approaches with an improvement of 2.5% in terms of Abs Rel and 1.1% in terms SILog against NeWCRFs.

[1] http://www.cvlibs.net/datasets/kitti/eval_depth.php?benchmark=depth_prediction

Method	Venue	Sq Rel↓	Abs Rel↓	RMSE↓	$\log_{10}$↓	δ_1↑	δ_2↑	δ_3↑
Eigen *et al.* [4]	NIPS'14	1.548	0.203	6.307	0.282	0.702	0.898	0.967
Godard *et al.* [9]	CVPR'17	0.898	0.114	4.935	0.206	0.861	0.960	0.976
Kuznietsov *et al.* [16]	CVPR'17	0.741	0.113	4.621	0.189	0.862	0.964	0.986
Gan *et al.* [6]	ECCV'18	0.666	0.098	3.933	0.173	0.890	0.984	0.985
DORN [5]	CVPR'18	0.307	0.072	2.727	0.120	0.932	0.984	0.994
Yin *et al.* [34]	ICCV'19	-	0.072	3.258	0.117	0.938	0.990	0.998
BTS [17]	Arxiv 19	0.245	0.059	2.756	0.096	0.956	0.993	0.998
PackNet-SAN* [10]	ICCV'21	-	0.062	2.888	-	0.955	-	-
TransDepth [33]	ICCV'21	0.252	0.064	2.755	0.098	0.956	0.994	0.994
Adabins [2]	CVPR'21	0.190	0.058	2.360	0.088	0.964	0.995	**0.999**
DPT* [25]	ICCV'21	-	0.060	2.573	0.092	0.959	0.995	0.996
Naderi *et al.* [21]	WACV'22		0.070	3.223	0.113	0.944	0.991	0.998
NeWCRFs [35]	CVPR'22	0.155	0.052	2.129	0.079	0.974	**0.997**	**0.999**
PixelFormer (ours)		**0.149**	**0.051**	**2.081**	**0.077**	**0.976**	**0.997**	**0.999**

Table 2: Results on KITTI Eigen Split test set [4]. The best results are in **bold** and second best are underlined. "*" means using additional data for training. ↑ means higher the better and ↓ means lower the better. An upper bound of 80 meters on the ground truth depth map is used for evaluation. All the numbers have been taken from the corresponding papers.

Method	Venue	Sq Rel↓	Abs Rel↓	RMSE↓	$\log_{10}$↓	δ_1↑	δ_2↑	δ_3↑
Chen *et al.* [3]	IJCAI'19	-	0.166	0.494	0.071	0.757	0.943	0.984
Yin *et al.* [34]	ICCV'19	-	0.183	0.541	0.082	0.696	0.912	0.973
BTS [17]	Arxiv'19	-	0.172	0.515	0.075	0.740	0.933	0.980
Adabins [2]	CVPR'21	-	0.159	0.476	0.068	0.771	0.944	0.983
PixelFormer (ours)		**0.0915**	**0.144**	**0.441**	**0.062**	**0.802**	**0.962**	**0.990**

Table 3: Results on SUNRGB-D test set without fine-tuning the models trained on NYUV2. The best results are in **bold** and second best are underlined. ↑ means higher the better and ↓ means lower the better. An upper bound of 8 meters on the ground truth depth map is used for evaluation. The numbers have been taken from the [2].

Results on SUNRGB-D: Following [2], we analyze our network's generalization performance by evaluating the model performance on test SUNRGB-D without finetuning the model on the NYUV2 dataset. As shown in Table 3, PixelFormer outperforms Adabins by 9.4% and 7.4% in terms of Abs Rel and RMSE, respectively. Thus demonstrating the effectiveness of pixel-adaptive global local fusion for out-of-distribution input images.

6. Ablation Study

Efficacy of Skip Attention Module: Table 4 demonstrates the effectiveness of our proposed SAM module against other baseline convolution-based alternatives *Add-Conv* and *Cat-Conv* to combine the encoder and decoder features at a particular scale. Add-Conv fuses the encoder-decoder features by pixelwise addition followed by a convolution operation. Cat-Conv concatenates the encoder and decoder features w.r.t. the channel dimension, followed by a convolution operation. The addition-based approach outperforms the concatenation approach by a small margin. However, using our SAM module outperforms Add-Conv by 4.0% in terms of Abs Rel and 4.2% in terms of Sq. Rel. This validates the contribution of the proposed SAM module.

Method	Abs Rel↓	Sq Rel↓	δ_1↑
Add-Conv	0.0602	0.190	0.964
Cat-Conv	0.0613	0.192	0.964
Decoder-Ours (SAM)	**0.0578**	**0.182**	**0.967**

Table 4: Ablation experiment to demonstrate the efficacy of SAM module on KITTI Eigen Split using Swin-T as the encoder. ↑ means higher the better and ↓ means lower the better. The best results are in **bold** and second best are underlined. An upper bound of 80 meters on the ground truth depth map is used for evaluation.

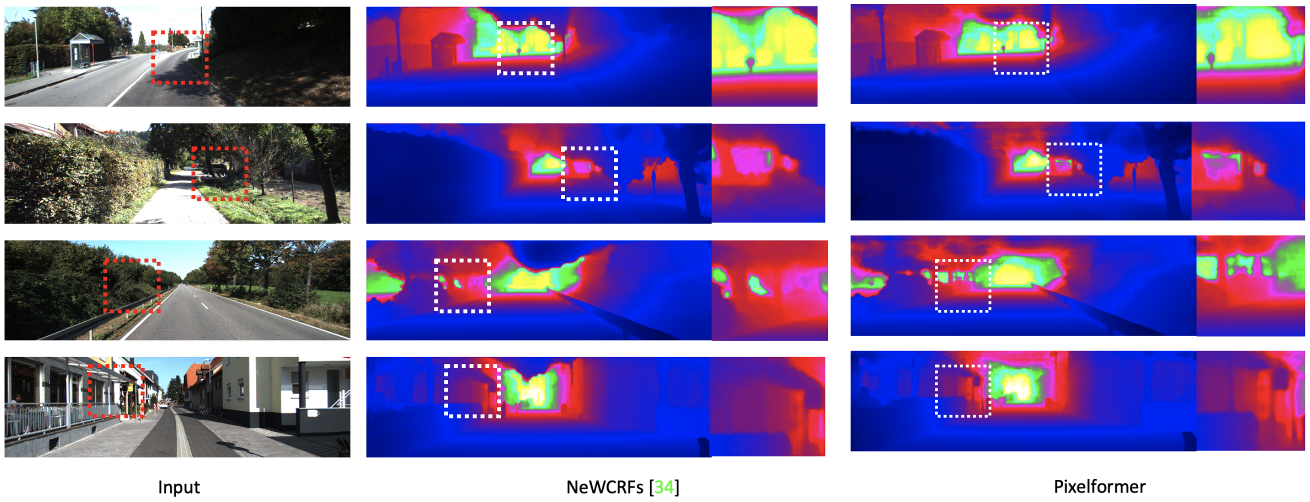

Figure 4: Qualitative comparison on the outdoor dataset KITTI. As shown in the top row, PixelFormer can precisely estimate depth maps for far-sighted road objects.

Method	Abs Rel↓	Sq Rel↓	δ_1↑
mViT-Last	0.0596	0.190	0.964
mViT-First	0.0584	0.185	0.966
Ours (BCP)	**0.0578**	**0.183**	**0.967**

Table 5: Ablation experiment to demonstrate the usefulness of embedding the depth information into the initial pixel queries on KITTI Eigen Split using Swin-T as the encoder. ↑ means higher the better and ↓ means lower the better. The best results are in **bold** and second best are underlined. An upper bound of 80 meters on the ground truth depth map is used for evaluation.

Effectiveness of embedding the depth information into the pixel queries: We experiment to showcase the usefulness of using the initial pixel queries to predict the bin centers. We compare our proposed design that predicts the bin centers using the initial pixel queries against using a ViT to predict the bin centers as in [2]. mViT-Last converts the feature map of the highest resolution (F shown in Fig. 2) into 16×16 patches and integrates the information in the first patch using multiple self-attention layers ($L = 4$). The first patch embedding is passed through MLP layers to predict the bin centers. mViT-First predicts bin centers by passing the initial pixel queries to a ViT. We use patch size = 1 for mViT-First for a fair comparison. Table 5 shows that both mVit-First and our approach outperform mVit-Last by 2.0% and 3.0%, respectively, in terms of Abs Rel, indicating that embedding depth information into the initial pixel queries via direct loss supervision helps predict better depth estimates. mVit-First does not give any further benefit to predicting bin centers since global information is already aggregated into the initial pixel queries via the PQI module.

7. Conclusion

This work presents PixelFormer, a novel encoder-decoder strategy for Monocular Depth Estimation that poses the problem as a pixel query refinement problem. The global initial pixel queries predicted by the Pixel Query Initialiser module are refined to a higher resolution by querying the multiscale encoder features at various resolutions through the proposed Skip Attention Module. Unlike convolution-based skip connections, the module can fuse decoder features with long-range dependency, leading to more accurate depth labels. Our proposed Bin Center Prediction module helps constrain the network with depth information embedded into the initial pixel queries through direct loss supervision. Through extensive experiments, we showcase that PixelFormer improves state-of-the-art performance on the indoor dataset NYUV2 and outdoor dataset KITTI by 5.3% and 3.9%, respectively, along with an improved generalization performance by 9.4% on the indoor SUNRGBD dataset. In the future, we will try to apply our content adaptive fusion using SAM to other dense estimation tasks like semantic segmentation.

Acknowledgement: This work has been partly supported by the funding received from DST through the IMPRINT program (IMP/2019/000250). We acknowledge National Supercomputing Mission (NSM) for providing computing resources of ‘PARAM Siddhi-AI’, under National PARAM Supercomputing Facility, CDAC Pune, and supported by the Ministry of Electronics and Information Technology and Department of Science and Technology, Govt. of India.

References

[1] Ashutosh Agarwal and Chetan Arora. Depthformer: Multiscale vision transformer for monocular depth estimation with global local information fusion. In *2022 IEEE International Conference on Image Processing (ICIP)*, pages 3873–3877, 2022. 1

[2] Shariq Farooq Bhat, Ibraheem Alhashim, and Peter Wonka. Adabins: Depth estimation using adaptive bins. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4009–4018, June 2021. 1, 2, 3, 4, 5, 6, 7, 8

[3] Xiaotian Chen, Xuejin Chen, and Zheng-Jun Zha. Structure-aware residual pyramid network for monocular depth estimation. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, IJCAI'19, page 694–700. AAAI Press, 2019. 7

[4] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014. 1, 2, 5, 6, 7

[5] Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, and Dacheng Tao. Deep ordinal regression network for monocular depth estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 1, 2, 3, 6, 7

[6] Yukang Gan, Xiangyu Xu, Wenxiu Sun, and Liang Lin. Monocular depth estimation with affinity, vertical pooling, and label enhancement. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision – ECCV 2018*, pages 232–247, Cham, 2018. Springer International Publishing. 7

[7] Ravi Garg, Vijay Kumar B.G., Gustavo Carneiro, and Ian Reid. Unsupervised cnn for single view depth estimation: Geometry to the rescue. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision – ECCV 2016*, pages 740–756, Cham, 2016. Springer International Publishing. 5

[8] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *International Journal of Robotics Research (IJRR)*, 2013. 5

[9] Clement Godard, Oisin Mac Aodha, and Gabriel J. Brostow. Unsupervised monocular depth estimation with left-right consistency. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 7

[10] Vitor Guizilini, Rares Ambrus, Wolfram Burgard, and Adrien Gaidon. Sparse auxiliary networks for unified monocular depth prediction and completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11078–11088, June 2021. 5, 6, 7

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 346–361, Cham, 2014. Springer International Publishing. 3

[12] Lam Huynh, Phong Nguyen-Ha, Jiri Matas, Esa Rahtu, and Janne Heikkilä. Guiding monocular depth estimation using depth-attention volume. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 581–597, Cham, 2020. Springer International Publishing. 6

[13] Nabil Ibtehaz and M. Sohel Rahman. Multiresunet : Rethinking the u-net architecture for multimodal biomedical image segmentation. *Neural Networks*, 121:74–87, 2020. 2

[14] Yasunori Ishii and Takayoshi Yamashita. Cutdepth: Edge-aware data augmentation in depth estimation. *ArXiv*, abs/2107.07684, 2021. 5

[15] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. 5

[16] Yevhen Kuznietsov, Jorg Stuckler, and Bastian Leibe. Semi-supervised deep learning for monocular depth map prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 7

[17] Jin Han Lee, Myung-Kyu Han, Dong Wook Ko, and Il Hong Suh. From big to small: Multi-scale local planar guidance for monocular depth estimation. *arXiv preprint arXiv:1907.10326*, 2019. 1, 5, 6, 7

[18] Minhyeok Lee, Sangwon Hwang, Chaewon Park, and Sangyoun Lee. Edgeconv with attention module for monocular depth estimation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2858–2867, January 2022. 1, 2, 6

[19] Tsung-Yi Lin, Piotr Dollar, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 1, 4

[20] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10012–10022, October 2021. 1, 2, 3, 4, 5

[21] Taher Naderi, Amir Sadovnik, Jason Hayward, and Hairong Qi. Monocular depth estimation with adaptive geometric attention. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 944–954, January 2022. 1, 2, 6, 7

[22] Ozan Oktay, Jo Schlemper, Loic Le Folgoc, Matthew Lee, Mattias Heinrich, Kazunari Misawa, Kensaku Mori, Steven McDonagh, Nils Y Hammerla, Bernhard Kainz, Ben Glocker, and Daniel Rueckert. Attention U-Net: Learning Where to Look for the Pancreas. *arXiv e-prints*, page arXiv:1804.03999, Apr. 2018. 2

[23] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison,

Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. 5

[24] Vaishakh Patil, Christos Sakaridis, Alexander Liniger, and Luc Van Gool. P3depth: Monocular depth estimation with a piecewise planarity prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1610–1621, June 2022. 2, 5, 6

[25] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 12179–12188, October 2021. 2, 6, 7

[26] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham, 2015. Springer International Publishing. 2

[27] Wenzhe Shi, Jose Caballero, Ferenc Huszar, Johannes Totz, Andrew P. Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 4

[28] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In Andrew Fitzgibbon, Svetlana Lazebnik, Pietro Perona, Yoichi Sato, and Cordelia Schmid, editors, *Computer Vision – ECCV 2012*, pages 746–760, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg. 5, 6

[29] M. Song, S. Lim, and W. Kim. Monocular depth estimation using laplacian pyramid-based depth residuals. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(11):4381–4393, Nov. 2021. 2

[30] Shuran Song, Samuel P. Lichtenberg, and Jianxiong Xiao. Sun rgb-d: A rgb-d scene understanding benchmark suite. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. 5

[31] Lijun Wang, Jianming Zhang, Yifan Wang, Huchuan Lu, and Xiang Ruan. Cliffnet for monocular depth estimation with hierarchical embedding loss. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 316–331, Cham, 2020. Springer International Publishing. 2

[32] Xin Wen, Tianyang Li, Zhizhong Han, and Yu-Shen Liu. Point cloud completion by skip-attention network with hierarchical folding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[33] Guanglei Yang, Hao Tang, Mingli Ding, Nicu Sebe, and Elisa Ricci. Transformer-based attention networks for continuous pixel-wise prediction. In *ICCV*, 2021. 1, 6, 7

[34] Wei Yin, Yifan Liu, Chunhua Shen, and Youliang Yan. Enforcing geometric constraints of virtual normal for depth prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019. 1, 2, 6, 7

[35] Weihao Yuan, Xiaodong Gu, Zuozhuo Dai, Siyu Zhu, and Ping Tan. Neural window fully-connected crfs for monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3916–3925, June 2022. 1, 2, 3, 4, 5, 6, 7

[36] Nikolaos Zioulis, Georgios Albanis, Petros Drakoulis, Federico Alvarez, Dimitrios Zarpalas, and Petros Daras. Hybrid skip: A biologically inspired skip connection for the UNet architecture. *IEEE Access*, 10:53928–53939, 2022. 2