DCDepth: Progressive Monocular Depth Estimation in Discrete Cosine Domain

Kun Wang[1], Zhiqiang Yan[1], Junkai Fan[1], Wanlu Zhu[1], Xiang Li[2], Jun Li[1]* and Jian Yang[1]*

[1]PCA Lab, Nanjing University of Science and Technology, China
[2]Nankai University, China

Abstract

In this paper, we introduce DCDepth, a novel framework for the long-standing monocular depth estimation task. Moving beyond conventional pixel-wise depth estimation in the spatial domain, our approach estimates the frequency coefficients of depth patches after transforming them into the discrete cosine domain. This unique formulation allows for the modeling of local depth correlations within each patch. Crucially, the frequency transformation segregates the depth information into various frequency components, with low-frequency components encapsulating the core scene structure and high-frequency components detailing the finer aspects. This decomposition forms the basis of our progressive strategy, which begins with the prediction of low-frequency components to establish a global scene context, followed by successive refinement of local details through the prediction of higher-frequency components. We conduct comprehensive experiments on NYU-Depth-V2, TOFDC, and KITTI datasets, and demonstrate the state-of-the-art performance of DCDepth. Code is available at `https://github.com/w2kun/DCDepth`.

1 Introduction

Monocular **D**epth **E**stimation (MDE) is a cornerstone topic within computer vision communities, tasked with predicting the distance–or depth–of each pixel's corresponding object from the camera based solely on single image. As a pivotal technology for interpreting 3D scenes from 2D representations, MDE is extensively applied across various fields such as autonomous driving, robotics, and 3D modeling [45, 49, 9, 43], *etc*. However, MDE is challenged by the inherent ill-posed nature of inferring 3D structures from 2D images, making it a particularly daunting task for traditional methodologies, which often hinge on particular physical assumptions or parametric models [40, 59, 31, 32].

Over the past decade, the field of computer vision has witnessed a substantial surge in the integration of deep learning techniques. Many studies have endeavored to harness the robust learning capabilities of end-to-end deep neural networks for MDE task, propelling the estimation accuracy to new heights. Researchers have investigated a variety of methodologies, including regression-based [11, 19, 55], classification-based [5, 12], and classification-regression based approaches [3, 20], to predict depth on a per-pixel basis within the spatial domain. Despite these significant strides in enhancing accuracy, current methods encounter two primary limitations: the first is the tendency to predict depth for individual pixels in isolation, thus neglecting the crucial local inter-pixel correlations. The second limitation is the reliance on a singular forward estimation process, which may not sufficiently capture the complexities of 3D scene structures, thereby constraining their predictive performance.

To address the identified limitations, we propose to transfer depth estimation from the spatial domain to the frequency domain. Instead of directly predicting metric depth values, our method focuses on estimating the frequency coefficients of depth patches transformed using the **D**iscrete **C**osine

---

*Corresponding authors.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).
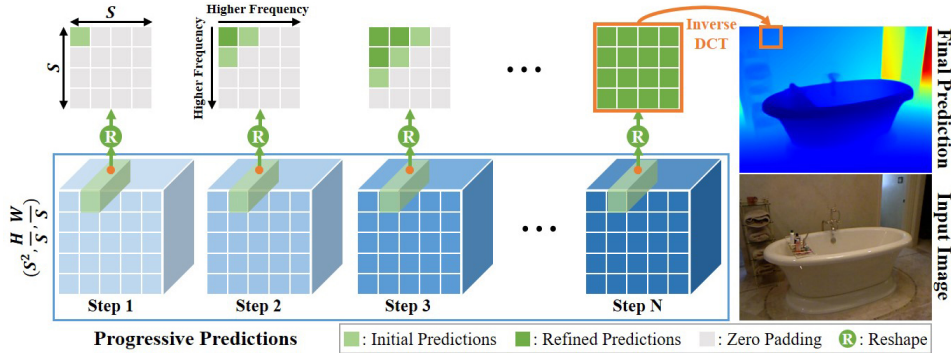
Figure 1: **Progressive estimation scheme.** For input image with size $H \times W$, DCDepth estimates the DCT coefficients for each $S \times S$ depth patches. The prediction follows a global-to-local strategy, starting with the initial estimation of lower-frequency components to capture the global scene structure. Subsequently, higher-frequency components are estimated to enhance the local details, while the lower-frequency estimates are refined. The estimation is carried out at $\frac{H}{S} \times \frac{W}{S}$ resolution, and spatial-domain estimation is achieved through inverse DCT.

Transform (DCT) [2, 6]. This strategy offers dual benefits: firstly, the DCT's basis functions inherently capture the inter-pixel correlations within depth patches, thereby facilitating the model's learning of local structures. Secondly, the DCT decomposes depth information into distinct frequency components, where low-frequency components reflect the overall scene architecture, and high-frequency components capture intricate local details. This dichotomy underpins our progressive estimation methodology, which commences with the prediction of low-frequency coefficients to grasp the macroscopic scene layout, subsequently refining the local geometries by inferring higher-frequency coefficients predicated on previous predictions. The spatial depth map is then accurately reconstructed via the inverse DCT. We illustrate this progress in Fig. 1. To implement our progressive estimation, we introduce a *Progressive Prediction Head* (PPH) that conditions on previous predictions from both spatial and frequency domains, and facilitates the sequential prediction of higher-frequency components using a GRU-based mechanism. Furthermore, recognizing the DCT's energy compaction property–indicative of the concentration of signal data within low-frequency components–we introduce a DCT-inspired downsampling technique to mitigate information loss during the downsampling process. This technique is embedded within a *Pyramid Feature Fusion* (PFF) module, ensuring effective fusion of multi-scale image features for accurate depth estimation.

Our contributions can be succinctly summarized in three key aspects:

- To the best of our knowledge, we are the first to formulate MDE as a progressive regression task in the discrete cosine domain. Our proposed method not only models local correlations effectively but also enables global-to-local depth estimation.

- We introduce a framework called DCDepth, comprising two novel modules: the PPH module progressively estimates higher-frequency coefficients based on previous predictions, and the PFF module incorporates a DCT-based downsampling technique to mitigate information loss during downsampling and ensures effective integration of multi-scale features.

- We evaluate our approach through comprehensive experiments on NYU-Depth-V2 [36], TOFDC [52], and KITTI [13] datasets. The results demonstrate the superior performance of DCDepth compared to existing state-of-the-art methods.

2   Related Work

Monocular **D**epth **E**stimation (MDE) remains a central theme in computer vision, essential for translating 2D imagery into 3D scene geometry. The evolution of MDE has been markedly influenced by the integration of deep neural networks. A foundational advancement was introduced by Eigen et al. [11], who developed a multi-scale deep convolutional network architecture, comprising a global network for coarse depth prediction and a local network for refinement. They also introduced a scale-invariant loss function to address the scale ambiguity challenge inherent in MDE. Building on

2

this, subsequent researches [19, 56, 42, 51, 50, 57, 53] have adopted end-to-end regression approaches with deep convolutional networks to further tackle MDE's challenges.

However, inferring depth from a single image is intrinsically problematic due to the countless potential depth maps that can correspond to one image. To mitigate this, additional information and constraints have been incorporated into the MDE task, such as semantics [44, 60] and surface normals [28, 33]. Further enhancements in depth estimation accuracy have been achieved through attention mechanisms [14, 47, 30], multivariate gaussian modeling [21], internal discretization technique [27] and pretraining [48, 46]. In contrast to the regression-based approach, some works [12, 5] have conceptualized MDE as a classification task, estimating the probability distribution of depth values. Yet, these methods often produce discontinuities due to discrete depth outputs. To overcome this, alternative strategies [3, 20, 4, 34] have combined classification and regression formulations, learning probabilistic distributions and employing linear combinations with depth candidates for final depth predictions. Our methodology diverges from these paradigms by progressively estimating frequency coefficients for depth patches after their transformation into the discrete cosine domain. This approach not only enhances computational efficiency but also achieves state-of-the-art performance.

3   Method

In this section, we introduce our progressive depth estimation framework, DCDepth. We begin by providing an overview of the 2D **D**iscrete **C**osine **T**ransform (DCT) as essential background knowledge. Subsequently, we delve into the progressive estimation scheme and elaborate on the network architecture. Finally, we present the loss function employed for training our model.

3.1   Reviewing 2D Discrete Cosine Transform

The 2D DCT is a mathematical technique used to decompose 2D discrete signals, such as depth maps and feature maps, into a sum of cosine basis functions with varying frequencies. The basis functions are defined as follows:

$$B_{u,v}^{i,j} = \alpha(u)\alpha(v) \cos\left[\frac{\pi}{W}\left(i + \frac{1}{2}\right)u\right] \cos\left[\frac{\pi}{H}\left(j + \frac{1}{2}\right)v\right], \tag{1}$$

where $u \in [0, W-1]$ and $v \in [0, H-1]$ represent the frequency indices, $i \in [0, W-1]$ and $j \in [0, H-1]$ denote the signal indices, and W and H indicate the input resolution. The terms $\alpha(u)$ and $\alpha(v)$ correspond to normalization factors. The forward process of 2D DCT, denoted as $T(\cdot)$, transforms the input signal $x \in \mathbb{R}^{H \times W}$ in the spatial domain to the frequency spectrum $f = T(x), f \in \mathbb{R}^{H \times W}$, and can be expressed as:

$$f_{u,v} = \sum_{i=0}^{W-1} \sum_{j=0}^{H-1} x_{i,j} B_{u,v}^{i,j}. \tag{2}$$

The resulting f is a matrix with the same size as the input x, with low-frequency components located near the top-left corner and high-frequency components near the bottom-right corner. The upper left one with zero frequency is called the DC components, and the remains are AC components. Low-frequency components typically characterize smooth regions, while high-frequency components capture edges or fine details where signal values change rapidly. The inverse 2D DCT, denoted as $T^{-1}(\cdot)$, performs the reverse operation by transforming the frequency spectrum f back to the spatial domain $x = T^{-1}(f)$, and can be formulated as:

$$x_{i,j} = \sum_{u=0}^{W-1} \sum_{v=0}^{H-1} f_{u,v} B_{u,v}^{i,j}. \tag{3}$$

The DCT has two desirable advantages. Firstly, it operates in the real number domain, simplifying the data processing. Secondly, it exhibits superior energy compaction properties by concentrating the majority of information within a small number of low-frequency components.

3.2   Progressive Estimation in Discrete Cosine Domain

Estimating depth from a single image remains a challenging task, particularly for scenes with intricate geometry. To tackle this, we propose a progressive method based on 2D DCT to estimate scene depth

3
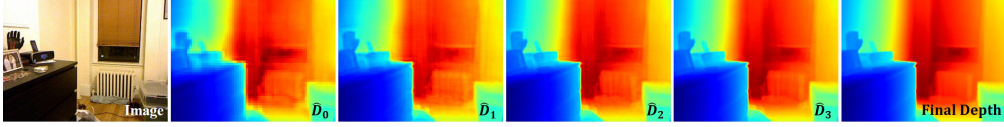
Figure 2: **Evolution of intermediate depth estimations.** We report several intermediate depth estimation results to illustrate our progressive estimation scheme.

progressively from a global perspective down to local details. The entire process is illustrated in Fig. 1. We denote the input image as $\mathcal{I} \in \mathbb{R}^{3 \times H \times W}$. Our proposed method, symbolized as $\Psi(\cdot)$, predicts the frequency coefficients $\mathcal{C} \in \mathbb{R}^{S^2 \times \frac{H}{S} \times \frac{W}{S}}$ for non-overlapping depth patches $\mathcal{P} \in \mathbb{R}^{S \times S}$, where S is set to 8 in our framework. These coefficients are subsequently transformed back to the spatial domain $\hat{\mathcal{D}} \in \mathbb{R}^{H \times W}$ using the inverse 2D DCT, as expressed by

$$\hat{\mathcal{D}} = T^{-1}(\Psi(\mathcal{I})). \tag{4}$$

The separation of low- and high-frequency components in a depth map effectively divides the scene into overall structures with gradual depth changes and local details with sharp depth transitions. This frequency characteristic enables us to break down the challenging MDE task into multiple prediction stages, progressing from simpler to more complex predictions. Initially, the DC coefficient $\mathcal{C}_0$ is predicted, establishing a foundational depth context. Subsequently, the AC coefficients $\{\mathcal{C}_i\}_{i=1}^{S^2-1}$ are iteratively estimated in ascending frequency order. During the inverse transformation to the spatial domain, any coefficients yet to be predicted are padded with zeros. In each iterative step k, we not only predict higher-frequency components but also refine the preceding frequency predictions

$$\mathcal{C}^k = \mathcal{C}^{k-1} + \Delta\mathcal{C}^k, \tag{5}$$

by estimating a correction term $\Delta\mathcal{C}^k$. To reduce the required iterations for estimating all S^2 coefficients, we utilize the energy compaction property of DCT, and partition the frequency spectrum $\mathcal{C}$ into subgroups along the subdiagonal, yielding $2S - 1$ subgroups $\{g_i\}_{i=0}^{2S-1}$. By merging the high-frequency subgroups, we further streamline the iterative process. This grouping strategy ensures that lower-frequency groups contain fewer components necessitating more prediction steps, while higher-frequency groups encompass a larger number of components requiring fewer steps. The intermediate depth maps are provided in Fig. 2 to elucidate the step-by-step prediction process.

3.3 DCDepth Architecture

Overview We present the comprehensive framework of DCDepth in Fig. 3, which comprise four key components: an image encoder, a **P**yramid **F**eature **F**usion (PFF) module, a decoder, and a **P**rogressive **P**rediction **H**ead (PPH). The image encoder acts as a robust feature extractor capturing image features $\mathcal{F} = \{\mathcal{F}_0, \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3\}$ at varying resolutions of $1/4$, $1/8$, $1/16$, and $1/32$ relative to the input image size. These multi-scale features are advantageous as the shallow features contain texture-related details, while the deep features hold global and semantic information essential for scene understanding. The PFF module, symbolized as $\Gamma(\cdot)$, is introduced to effectively amalgamate these features, yielding a comprehensive integrated feature representation $\mathcal{F}' = \Gamma(\mathcal{F})$. The decoder, denoted as $D(\cdot)$, consists of three neural CRF [58] modules and two PixelShuffle [35] modules. This configuration processes and upscales $\mathcal{F}'$ to $\hat{\mathcal{F}} = D(\mathcal{F}')$, achieving $1/8$ of the original resolution. The PPH performs estimations at the same resolution as $\hat{\mathcal{F}}$. It begins by down-sampling $\mathcal{F}_0$ to half its resolution using the proposed DCT-based downsampling. This down-sampled feature is then concatenated with $\hat{\mathcal{F}}$, forming the initial hidden state for the progressive estimation.

Pyramid Feature Fusion Module The primary objective of PFF is to harness the wealth of information embedded in the multi-scale image features, thereby creating a more comprehensive and enriched feature representation conducive to scene understanding. The layout of PFF is depicted in the left box of Fig. 3. Effective feature aggregation necessitates a proficient downsampling strategy to mitigate information loss, especially when downscaling at larger magnifications. To address this, we introduce a novel DCT-based downsampling strategy engineered to minimize information loss during downsampling. The operational procedure of this strategy is elucidated in the bottom-left corner of Fig. 3. Consider a feature map $\mathcal{F} \in \mathbb{R}^{C \times H \times W}$ slated for downsampling by a factor of R. We begin

4
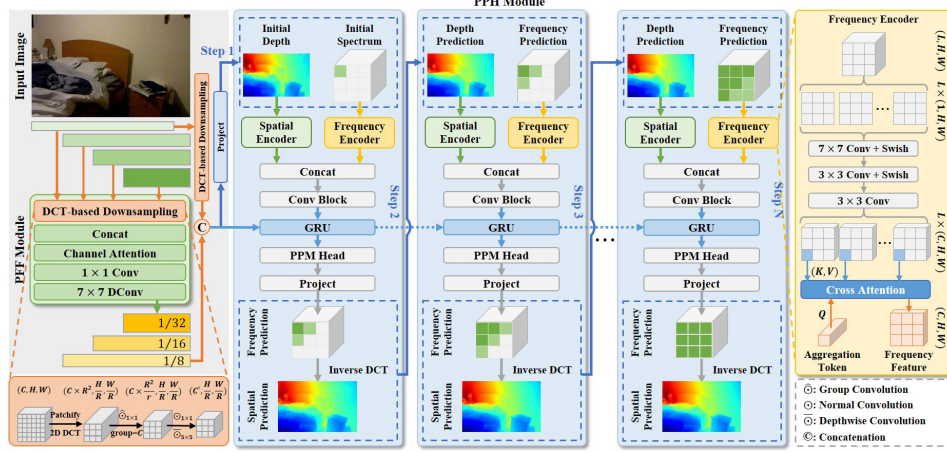
Figure 3: DCDepth framework overview. The DCT-based downsampling strategy is shown at the bottom-left corner, where R and r denote for downsampling factor and channel reduction rate, respectively. The central section details the iterative process of PPH, with N indicating the number of iterative steps. The frequency encoder utilized by PPH is illustrated at the right box.

by partitioning $\mathcal{F}$ into patches $\mathcal{P} \in \mathbb{R}^{C \times R^2 \times \frac{H}{R} \times \frac{W}{R}}$. Each channel of $\mathcal{P}$ is then individually subjected to Eq. 2 to transform the feature maps into the frequency domain. Leveraging the energy compaction property of the DCT, the key information within $\mathcal{F}$ is condensed into a few dominant frequency components characterized by large absolute values. This compression enables us to selectively reduce the number of channels from $C \times R^2$ to $C \times \frac{R^2}{r}$ with a reduction rate of r via 1×1 convolutions configured with groups set to C. The squeezed feature maps are then consolidated through a sequence of operations involving a 1×1 convolution followed by a 5×5 depth-wise convolution, culminating in the generation of the final output featuring C' channels and reduced spatial resolution.

Progressive Prediction Head   The PPH, as depicted in the middle segment of Fig. 3, incorporates two specialized encoders: $E_s(\cdot)$ for spatial-domain inputs and $E_f(\cdot)$ for frequency-domain inputs. The spatial encoder, composed of three convolutional layers with a stride of 2, convolves and downsamples the spatial-domain input $\hat{\mathcal{D}}$, producing a feature map at $1/8$ of the original resolution. The architecture of $E_f(\cdot)$ is outlined in the right box of Fig. 3. For frequency input $\mathcal{C} \in \mathbb{R}^{L \times H \times W}$, where L signifies the number of valid frequency components, we first split them into L chunks with shape $1 \times H \times W$. Each chunk is then processed through three convolutional layers with Swish activation [29] to extract features of dimensions $C \times H \times W$ for each frequency component. Given the variability in the number of valid frequency components across different iterative steps, we employ cross-attention [41, 10] mechanism to merge information from the various frequency components. A learnable *aggregation token* of dimensions $1 \times C$ is introduced to compile information from individual frequency components at each pixel location, yielding feature outputs of shape $C \times H \times W$ and effectively compressing the dimension L. The PPH operates iteratively, utilizing a **G**ated **R**ecurrent **U**nit (GRU) [7, 39], denoted as $G(\cdot, \cdot)$, to encode the historical estimation states

$$\mathcal{H}_i = G(E_s(\hat{\mathcal{D}}_{i-1}), E_f(\mathcal{C}_{i-1})), \tag{6}$$

prior to iterative step i. The hidden state $\mathcal{H}$ is then projected to the coefficient output by a Pyramid Pooling Module (PPM) [61] to aggregate global context, followed by a linear projection.

3.4   Loss Function

We employ the scaled scale-invariant loss [17, 3] to calibrate the model's depth estimations $\hat{\mathcal{D}}_i$ at each iterative step i against the ground truth depth map $\mathcal{D}$. The loss function is formulated as:

$$L_d = \alpha \cdot \sum_{i=1}^{N} \beta^{N-i} \sqrt{\frac{1}{M} \sum d_i^2 - \frac{\lambda}{M^2} (\sum d_i)^2}, \tag{7}$$

where $d = \hat{\mathcal{D}}_i - \mathcal{D}$, N denotes the number of iterative steps, and M represents the number of valid depth values. We consistently set $\alpha = 10$, $\beta = 0.8$ and $\lambda = 0.85$ across all experiments. The presence

5

Method	Backbone	Abs Rel ↓	Sq Rel ↓	RMSE ↓	$\log_{10}$ ↓	$\delta < 1.25$ ↑	$\delta < 1.25^2$ ↑	$\delta < 1.25^3$ ↑
DORN [12]	ResNet-101	0.115	–	0.509	0.051	0.828	0.965	0.992
VNL [55]	ResNet-101	0.108	–	0.416	0.048	0.875	0.976	0.994
BTS [17]	DenseNet-161	0.110	0.066	0.392	0.047	0.885	0.978	0.994
ASNDepth [24]	HRNet-48	0.101	–	0.377	0.044	0.890	0.982	0.996
TransDepth [54]	R-50+ViT-B/16	0.106	–	0.365	0.045	0.900	0.983	0.996
AdaBins [3]	E-B5+mini-ViT	0.103	–	0.364	0.044	0.903	0.984	<u>0.997</u>
LocalBins [4]	E-B5	0.099	–	0.357	0.042	0.907	0.987	**0.998**
NeWCRFS [58]	Swin-Large	0.095	0.045	0.334	0.041	0.922	**0.992**	**0.998**
BinsFormer [20]	Swin-Large	0.094	–	0.330	0.040	0.925	0.989	<u>0.997</u>
PixelFormer [1]	Swin-Large	0.090	–	0.322	0.039	0.929	<u>0.991</u>	**0.998**
IEBins [34]	Swin-Large	0.087	<u>0.040</u>	0.314	<u>0.038</u>	0.936	**0.992**	**0.998**
MG-Depth [21]	Swin-Large	0.087	–	<u>0.311</u>	–	0.933	–	–
NDDepth [33]	Swin-Large	0.087	0041	<u>0.311</u>	<u>0.038</u>	0.936	<u>0.991</u>	**0.998**
VA-DepthNet [22]	Swin-Large	<u>0.086</u>	**0.039**	**0.304**	**0.037**	<u>0.937</u>	**0.992**	**0.998**
Ours	Swin-Large	**0.085**	**0.039**	**0.304**	**0.037**	**0.940**	**0.992**	**0.998**

Table 1: **Quantitative depth comparison on NYU-Depth-V2 dataset.** The maximum depth is capped at 10 meters. R-50 and E-B5 represent for ResNet-50 [15] and EfficientNet-B5 [38], respectively. '-' means not applicable. The best result is in **bold**, and the second is <u>underlined</u>.
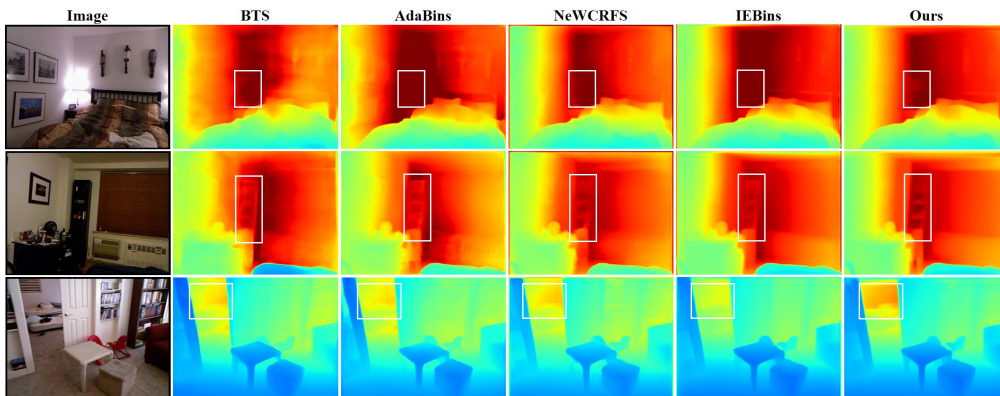
Figure 4: **Qualitative depth comparison on the NYU-Depth-V2 dataset.** The white boxes highlight the regions where our method achieves more accurate predictions.

of missing values in the depth ground truth can render the model's frequency-domain predictions inadequately supervised. To mitigate this, we introduce two regularization terms. Specifically, to enforce the sparsity of high-frequency coefficients, we define the frequency regularization loss as:

$$L_f = \sum (\epsilon^{u+v} - 1) \cdot |f_{u,v}|, \tag{8}$$

where $f_{u,v}$ is the frequency coefficient indexed by (u, v), and ϵ is set to 1.2. Additionally, we incorporate a smoothness term to promote the smoothness of $\hat{\mathcal{D}}$:

$$L_s = |\partial_x \hat{\mathcal{D}}| \cdot e^{-|\partial_x I_t|} + |\partial_y \hat{\mathcal{D}}| \cdot e^{-|\partial_y I_t|}, \tag{9}$$

where ∂_x and ∂_y represent image gradient along horizontal and vertical axes, respectively, and $|\cdot|$ denote the absolute value function. The final loss is the weighted summation of these three loss terms.

4 Experiment

In this section, we evaluate DCDepth by conducting a comparative analysis with established methodologies. We commence by delineating the datasets and evaluation metrics employed in our evaluation. Subsequently, we detail the implementation specifics that underpin our experiments. Concluding this section, we demonstrate the efficacy of the proposed modules via extensive ablation studies.

4.1 Dataset and Evaluation Metric

Dataset We evaluate our method on three datasets that covers a diverse array of indoor and outdoor scenes. **(1) NYU-Depth-V2** [36] is centered on indoor environments and consists of RGB-D images

6

Method	Backbone	Abs Rel ↓	Sq Rel ↓	RMSE ↓	RMSE log ↓	$\delta < 1.25$ ↑	$\delta < 1.25^2$ ↑	$\delta < 1.25^3$ ↑
BTS [17]	DenseNet-161	0.407	0.082	0.998	0.567	0.985	<u>0.998</u>	**1.000**
AdaBins [3]	E-B5+mini-ViT	0.279	0.044	0.729	0.462	0.990	<u>0.998</u>	**1.000**
NeWCRFS [58]	Swin-Large	0.533	0.244	1.004	0.792	0.956	0.976	0.988
PixelFormer [1]	Swin-Large	0.534	0.230	1.076	0.782	0.957	0.979	<u>0.991</u>
VA-DepthNet [22]	Swin-Large	<u>0.234</u>	<u>0.029</u>	<u>0.619</u>	<u>0.373</u>	**0.996**	**0.999**	**1.000**
IEBins [34]	Swin-Large	0.528	0.238	0.999	0.790	0.956	0.976	0.988
Ours	Swin-Large	**0.188**	**0.027**	**0.565**	**0.352**	<u>0.995</u>	**0.999**	**1.000**

Table 2: **Quantitative depth comparison on TOFDC dataset.** The maximum depth is capped at 5 meters. The first four error metrics are multiplied by 10 for presentation.
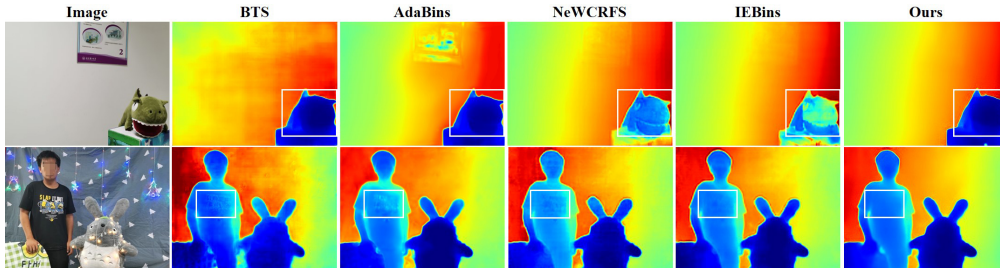
Figure 5: **Qualitative depth comparison on the TOFDC dataset.**

captured with a Microsoft Kinect sensor. The settings span various indoor scenes such as bedrooms, offices, and classrooms. The images in this dataset are presented at a resolution of 640×480. We follow the data split as outlined in BTS [17], featuring 24231 training images and 654 test images. **(2) TOFDC** [52] is collected using a mobile phone paired with a lightweight Time-of-Flight (ToF) camera, capturing a wide array of subjects like flowers, human figures, and toys under different scenes and lighting conditions. The dataset is divided into 10,000 training samples and 560 testing samples, with images at a resolution of 512×384. **(3) KITTI** [13] is a well-known outdoor dataset that features RGB images coupled with sparse depth maps obtained from a laser scanner mounted on a car. The images in this dataset have a resolution of 1216×352. We utilize both the Eigen split [11] and the official split for our analysis. The Eigen split comprises 23158 training images and 697 test images, while the official split includes 42949 training images and 500 test images.

Metrics   Consistent with prior works [58, 3, 34], we utilize a selection of well-established metrics to provide a comprehensive evaluation. The key metrics include: relative absolute error (Abs Rel), relative squared error (Sq Rel), root mean squared error (RMSE), absolute logarithmic error ($\log_{10}$), root mean squared logarithmic error (RMSE log), inverse root mean squared error (iRMSE) and threshold accuracy ($\delta < 1.25$, $\delta < 1.25^2$, and $\delta < 1.25^3$). Please refer to the appendix for details.

4.2   Implementation Detail

The DCDepth is implemented using Pytorch library [25], and is trained with a batch size of 8 on four NVIDIA RTX-4090 GPUs with data-distributed parallel computing. Our method is trained on NYU-Depth-V2 dataset for 20 epochs, TOFDC dataset for 25 epochs, KITTI eigen split for 20 epochs and KITTI official split for 12 epochs. The optimization objective of our method is a combination of the scale-invariant log loss L_d, the frequency regularization L_f and the smoothness regularization L_s, weighted by two scalar weights α and β:

$$L = L_d + \alpha \cdot L_f + \beta \cdot L_s. \tag{10}$$

For the NYU-Depth-V2 and TOFDC datasets, these two weights are set to 2×10^{-3} and 0.0, respectively, while for the KITTI dataset, both weights are set to 5×10^{-3}. We opt for the Adam optimizer [16] and leverage the OneCycle learning rate scheduler [37]. The learning rate schedule entails an initial increase from 2×10^{-5} to 10^{-4} during the first 2 epochs, followed by a subsequent decrease to 5×10^{-6} using a cosine annealing strategy. To enhance generalization and mitigate overfitting, we integrate various data augmentation techniques into the training pipeline, including random horizontal flips, random rotations, random color jitter, and random image filtering. For feature extraction from images, we incorporate a Swin-Transformer architecture [23] pretrained on the ImageNet dataset [8] as the image encoder. To reduce the iteration steps necessitated for spectrum

7

Method	Backbone	Abs Rel ↓	Sq Rel ↓	RMSE ↓	RMSE log ↓	$\delta < 1.25$ ↑	$\delta < 1.25^2$ ↑	$\delta < 1.25^3$ ↑
DORN [12]	ResNet-101	0.072	0.307	2.727	0.120	0.932	0.984	0.994
VNL [55]	ResNet-101	0.072	–	3.258	0.117	0.938	0.990	<u>0.998</u>
BTS [17]	DenseNet-161	0.060	0.249	2.798	0.096	0.955	0.993	<u>0.998</u>
TransDepth [54]	R-50+ViT-B/16	0.064	0.252	2.755	0.098	0.956	0.994	**0.999**
AdaBins [3]	E-B5+mini-ViT	0.058	0.190	2.360	0.088	0.964	<u>0.995</u>	**0.999**
P3Depth [26]	ResNet-101	0.071	0.270	2.842	0.103	0.953	0.993	<u>0.998</u>
NeWCRFS [58]	Swin-Large	0.052	0.155	2.129	0.079	0.974	**0.997**	**0.999**
BinsFormer [20]	Swin-Large	0.052	0.151	2.096	0.079	0.974	**0.997**	**0.999**
PixelFormer [1]	Swin-Large	<u>0.051</u>	0.149	2.081	<u>0.077</u>	<u>0.976</u>	**0.997**	**0.999**
VA-DepthNet [22]	Swin-Large	**0.050**	<u>0.148</u>	2.093	**0.076**	**0.977**	**0.997**	**0.999**
iDisc [27]	Swin-Large	**0.050**	**0.145**	<u>2.067</u>	<u>0.077</u>	**0.977**	**0.997**	**0.999**
Ours	Swin-Large	<u>0.051</u>	**0.145**	2.044	**0.076**	**0.977**	**0.997**	**0.999**

Table 3: **Quantitative depth comparison on the Eigen split of KITTI dataset.** The maximum depth value is capped at 80 meters.
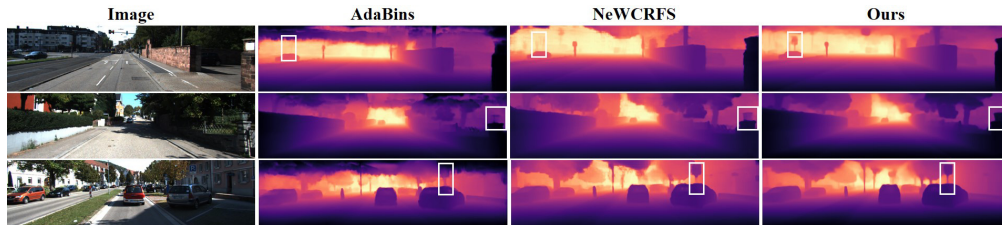
Figure 6: **Qualitative depth comparison on the Eigen split of KITTI dataset.**

prediction, we further merge the frequency subgroups with indices $\{6, 7\}$ and $\{8, \dots, 14\}$, leading to 9 iterative steps in total to generate the final depth predictions.

4.3 Comparison with the State-of-the-Art

NYU-Depth-V2 We benchmark our method against current **S**tate-**o**f-**T**he-**A**rt (SoTA) approaches on the indoor NYU-Depth-V2 dataset, with quantitative results presented in Tab. 1. Despite vision transformers elevating the precision of depth estimation on this dataset, our method has surpassed existing SoTA approaches, particularly in the *Abs Rel* and $\delta < 1.25$ metrics. Qualitative comparisons, illustrated in Fig. 4, reveal the adeptness of our method at capturing fine-grained geometries and producing smoother depth estimations in planar areas. Regions where our method outperforms are highlighted with white boxes, emphasizing its superior depth estimation accuracy.

TOFDC The TOFDC dataset is characterized by its dense ground truth depth data. By utilizing this dataset, we demonstrate the enhanced capability of our method to effectively harness the dense ground truth, thereby achieving more accurate depth estimations compared to existing SoTAs. We present the quantitative results in Tab. 2, where our method demonstrates superior performance over existing SoTAs across a majority of the evaluated metrics. Specifically, our method achieves a significant improvement on the *Abs Rel* and *RMSE* metrics compared to VA-DepthNet, with enhancements of 19.7% and 8.7%, respectively. Fig. 5 provides qualitative comparisons, illustrating that our method not only produces more accurate depth estimations but also more effectively delineates the object from the background, leading to more coherent depth estimations.

KITTI We further evaluate our method on the outdoor dataset, KITTI, which has sparse depth ground truth collected with LiDAR. This sparsity presents a contrast to the denser depth information available in the NYU and TOFDC datasets, resulting in less robust supervision for learning frequency coefficients. Despite this challenge, our method demonstrates its robustness by achieving SoTA performance, which is attributed to the utilization of plenty training data coupled with our proposed regularization constraints. The quantitative analysis, as detailed in Tab. 3, demonstrates the superior performance of our method. Qualitative evaluations, depicted in Fig. 6, further substantiate the superiority of our method. The quantitative results on KITTI official split are reported in Tab. 4. The pretrained weights from Semantic-SAM [18] are employed to initialize the encoder. Our method surpasses the compared approaches on the majority of metrics, particularly in the iRMSE metric, underscoring the robustness and effectiveness of our approach.

8

Metric	DORN [12]	BTS [17]	NeWCRFS [58]	PixelFormer [1]	BinsFormer [20]	iDisc [27]	VA-DepthNet [22]	IEBins [34]	NDDepth [33]	**Ours**
SILog ↓	11.77	11.67	10.39	10.28	10.14	9.89	9.84	9.63	<u>9.62</u>	**9.60**
Abs Rel ↓	8.78	9.04	8.37	8.16	8.23	8.11	7.96	<u>7.82</u>	**7.75**	7.83
Sq Rel ↓	2.23	2.21	1.83	1.82	1.69	1.77	1.66	1.60	<u>1.59</u>	**1.54**
iRMSE ↓	12.98	12.23	11.03	10.84	10.90	10.73	10.44	10.68	<u>10.62</u>	**10.12**

Table 4: **Quantitative depth comparison on the official split of KITTI dataset.** All metrics reported here are from the KITTI online leaderboard.

	NeWCRFS [58]	MG-Depth [21]	IEBins [34]	VA-DepthNet [22]	**Ours**				
					1 Step	2 Steps	3 Steps	4 Steps	9 Steps
Param (M) ↓	270	296	273	<u>262</u>	**259**				
Speed (FPS) ↑	**37.95**	24.24	21.51	15.68	<u>31.55</u>	28.72	26.03	24.07	14.24
RMSE ↓	0.334	0.311	0.314	**0.304**	0.310	0.307	0.306	<u>0.305</u>	**0.304**
δ < 1.25 ↑	0.922	0.933	0.936	0.937	0.937	<u>0.939</u>	<u>0.939</u>	<u>0.939</u>	**0.940**

Table 5: **Parameter efficiency and inference speed on NYU-Depth-v2 dataset.** The right section enumerates the inference speed and corresponding performance metrics of our method at various iteration stages. All models are benchmarked on a single RTX 4090 GPU for consistency.

Parameter efficiency We compare the parameter efficiency of our method with current SoTAs on the NYU-Depth-V2 dataset, with the input resolution set to 640×480. The quantitative results, presented in Tab. 5, reveal that our method exhibits the fewest training parameters while simultaneously achieving the best performance. For instance, our approach demonstrates a 9.0% improvement in the *RMSE* metric, while utilizing 4.1% fewer parameters than NeWCRFS.

4.4 Ablation Study

We conduct comprehensive ablation studies to demonstrate the efficacy of the proposed PPH and PFF modules, and analyze the impact of the iteration steps on both model performance and inference speed. All experiments presented in this section are conducted on the NYU-Depth-V2 dataset.

Effect of PPH module To assess the impact of the PPH module, we build a baseline by excluding the PPH from our method. In this setup, we employ a convolutional head to project the last-layer features to the output dimension. The final depth prediction is obtained through either bilinear and PixelShuffle [35] upsampling or inverse DCT that converts the predicted frequency coefficients back to the spatial domain. Additionally, we introduce the adaptive bins [3] as an alternative competitor. Quantitative experimental results are reported in Tab. 6. Among the three approaches outputting in the spatial domain, the PixelShuffle-based approach performs best. When predicting depth in the frequency domain, performance further improves, demonstrating the superiority of frequency-domain depth prediction. Lastly, our progressive prediction scheme significantly outperforms the compared approaches by a large margin, underscoring the efficacy of the PPH module.

Effect of PFF module To evaluate the impact of the PFF module, we establish a baseline by excluding the PFF component from our method. We first introduce a convolutional layer and a PPM [61] module to process the image feature at the last scale. Then, to validate the proposed DCT-based downsampling strategy, we replace it with bilinear and PixelUnshuffle [35] downsampling. The quantitative experimental results are reported in Tab. 7. The first two approaches, which only process the last-scale feature, perform worse than the competitors with multi-scale feature aggregation. This demonstrates the necessity of multi-scale feature aggregation for depth prediction. Furthermore, our method, employing the DCT-based downsampling strategy, achieves the best performance, showcasing the effectiveness of our proposed DCT-based strategy for feature downsampling.

Effect of iterative steps We analyze the impact of iterative steps on both prediction accuracy and inference speed. The results are reported in Tab. 5 and illustrated in Fig. 7. In summary, we observe that both prediction accuracy and inference time increase as the number of iterations grows. Leveraging the energy compaction property of the DCT, we strike a balance between accuracy and speed by selectively discarding predictions for high-frequency components. This strategic approach allows us to effectively reduce the number of iterative steps.

9

Method	Output Domain	Abs Rel ↓	Sq Rel ↓	RMSE ↓	$\delta < 1.25$ ↑	$\delta < 1.25^2$ ↑	$\delta < 1.25^3$ ↑
Baseline + Conv + Bilinear	Spatial-Domain	0.090	0.042	0.319	0.929	0.991	**0.998**
Baseline + AdaBins + Bilinear	Spatial-Domain	0.088	0.042	0.319	0.932	0.991	**0.998**
Baseline + Conv + PixelShuffle	Spatial-Domain	0.088	0.041	0.318	0.933	**0.992**	0.998
Baseline + Conv + inv DCT	Frequency-Domain	0.088	0.041	0.315	0.932	**0.992**	0.998
Baseline + PPH	Frequency-Domain	**0.085**	**0.039**	**0.304**	**0.940**	**0.992**	**0.998**

Table 6: **Ablation study on the PPH module.** The baseline is built by removing the PPH module. *Conv* denotes linear projection with a convolutional layer. *AdaBins* refers to the adaptive bins [3]. All methods output at $1/8$ scale, and Bilinear and PixelShuffle [35] are used to upsample the prediction.

Method	Abs Rel ↓	RMSE ↓	$\delta < 1.25$ ↑
Baseline + Conv	0.086	0.309	0.936
Baseline + PPM	0.086	0.306	0.939
Baseline + PFF (Bilinear)	**0.085**	0.305	**0.940**
Baseline + PFF (PixelUnshuffle)	**0.085**	0.306	**0.940**
Ours	**0.085**	**0.304**	**0.940**

Table 7: **Ablation study on the PFF module.** The baseline is built by removing the PFF module. We evaluate the proposed DCT-based downsampling strategy by replacing it with bilinear and PixelUnshuffle [35] downsampling.
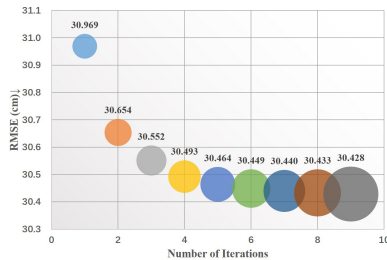
Figure 7: **Accuracy vs. inference speed.** The width of each bubble corresponds to the processing time.

5 Limitation and Broader Impact

Our method employs the differentiable inverse DCT to transform the predicted spectrum back to the spatial domain. By minimizing the difference between the spatial-domain estimation and the valid ground truth, our model can be trained end-to-end. However, the sparsity of the ground truth may lead to inefficient supervision of the frequency estimation. While we have proposed two regularization terms to prevent our model from being incorrectly optimized, we observe that our method is more effective with dense supervision. Exploring more effective training strategies when only sparse depth ground truth is available will be an important research direction for our future work.

Monocular depth estimation is a pivotal technique for interpreting 3D scenes from 2D images and has widespread applications in autonomous driving, robotics, and 3D modeling, among others. Given the extensive applications of this task, our method is poised to positively impact these fields by advancing their capabilities. Considering the fundamental nature of monocular depth estimation, our work is not anticipated to have a significant negative societal impact.

6 Conclusion

In this paper, we introduce DCDepth, a novel framework for the MDE task. Departing from existing methods, our method progressively estimates patch-wise depth in the frequency domain and then recovers spatial-domain depth via inverse DCT. This formulation inherently models local depth correlations and frames the estimation process as a global-to-local scheme, achieving more accurate depth estimation. Leveraging the energy compaction property of DCT, our method strikes an effective balance between accuracy and inference speed, making it well-suited for practical applications.

7 Acknowledgment

We would like to thank the reviewers and the chairs for their suggestions and efforts. This work was partially supported by the National Natural Science Foundation of China under Grant 62361166670 and 62072242, the Fundamental Research Funds for the Central Universities under Grant 070-63233084, the Young Scientists Fund of the National Natural Science Foundation of China under Grant 62206134 and the Tianjin Key Laboratory of Visual Computing and Intelligent Perception. The PCA Lab is associated with the Key Lab of Intelligent Perception and Systems for High-Dimensional Information of Ministry of Education, and Jiangsu Key Lab of Image and Video Understanding for Social Security, School of Computer Science and Engineering, Nanjing University of Sci & Tech.

10

References

[1] Ashutosh Agarwal and Chetan Arora. Attention attention everywhere: Monocular depth prediction with skip attention. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 5861–5870, 2023.

[2] N. Ahmed, T. Natarajan, and K.R. Rao. Discrete cosine transform. *IEEE Transactions on Computers*, C-23(1):90–93, 1974.

[3] 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*, pages 4009–4018, 2021.

[4] Shariq Farooq Bhat, Ibraheem Alhashim, and Peter Wonka. Localbins: Improving depth estimation by learning local distributions. In *European Conference on Computer Vision*, pages 480–496. Springer, 2022.

[5] Yuanzhouhan Cao, Zifeng Wu, and Chunhua Shen. Estimating depth from monocular images as classification using deep fully convolutional residual networks. *IEEE Transactions on Circuits and Systems for Video Technology*, 28(11):3174–3182, 2017.

[6] Wen-Hsiung Chen, C. Smith, and S. Fralick. A fast computational algorithm for the discrete cosine transform. *IEEE Transactions on Communications*, 25(9):1004–1009, 1977.

[7] Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar, 2014. Association for Computational Linguistics.

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[9] Xingshuai Dong, Matthew A. Garratt, Sreenatha G. Anavatti, and Hussein A. Abbass. Towards real-time monocular depth estimation for robotics: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 23(10):16940–16961, 2022.

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[11] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2014.

[12] 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*, pages 2002–2011, 2018.

[13] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3354–3361. IEEE, 2012.

[14] Zhixiang Hao, Yu Li, Shaodi You, and Feng Lu. Detail preserving depth estimation from a single image using attention guided networks. In *2018 International Conference on 3D Vision (3DV)*, pages 304–313. IEEE, 2018.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

11

[16] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[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.

[18] Feng Li, Hao Zhang, Peize Sun, Xueyan Zou, Shilong Liu, Jianwei Yang, Chunyuan Li, Lei Zhang, and Jianfeng Gao. Semantic-sam: Segment and recognize anything at any granularity. *arXiv preprint arXiv:2307.04767*, 2023.

[19] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2041–2050, 2018.

[20] Zhenyu Li, Xuyang Wang, Xianming Liu, and Junjun Jiang. Binsformer: Revisiting adaptive bins for monocular depth estimation. *IEEE Transactions on Image Processing*, 33:3964–3976, 2024.

[21] Ce Liu, Suryansh Kumar, Shuhang Gu, Radu Timofte, and Luc Van Gool. Single image depth prediction made better: A multivariate gaussian take. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17346–17356, 2023.

[22] Ce Liu, Suryansh Kumar, Shuhang Gu, Radu Timofte, and Luc Van Gool. Va-depthnet: A variational approach to single image depth prediction. *arXiv preprint arXiv:2302.06556*, 2023.

[23] 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*, pages 10012–10022, 2021.

[24] Xiaoxiao Long, Cheng Lin, Lingjie Liu, Wei Li, Christian Theobalt, Ruigang Yang, and Wenping Wang. Adaptive surface normal constraint for depth estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 12849–12858, 2021.

[25] 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*. Curran Associates, Inc., 2019.

[26] 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*, pages 1610–1621, 2022.

[27] Luigi Piccinelli, Christos Sakaridis, and Fisher Yu. idisc: Internal discretization for monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21477–21487, 2023.

[28] Xiaojuan Qi, Renjie Liao, Zhengzhe Liu, Raquel Urtasun, and Jiaya Jia. Geonet: Geometric neural network for joint depth and surface normal estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 283–291, 2018.

[29] Prajit Ramachandran, Barret Zoph, and Quoc V Le. Searching for activation functions. *arXiv preprint arXiv:1710.05941*, 2017.

[30] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 12179–12188, 2021.

[31] Ashutosh Saxena, Sung Chung, and Andrew Ng. Learning depth from single monocular images. In *Advances in Neural Information Processing Systems*. MIT Press, 2005.

[32] Ashutosh Saxena, Jamie Schulte, Andrew Y Ng, et al. Depth estimation using monocular and stereo cues. In *IJCAI*, pages 2197–2203, 2007.

12

[33] Shuwei Shao, Zhongcai Pei, Weihai Chen, Xingming Wu, and Zhengguo Li. Nddepth: Normal-distance assisted monocular depth estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7931–7940, 2023.

[34] Shuwei Shao, Zhongcai Pei, Xingming Wu, Zhong Liu, Weihai Chen, and Zhengguo Li. Iebins: Iterative elastic bins for monocular depth estimation. In *Advances in Neural Information Processing Systems*, pages 53025–53037. Curran Associates, Inc., 2023.

[35] Wenzhe Shi, Jose Caballero, Ferenc Huszár, 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*, pages 1874–1883, 2016.

[36] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part V 12*, pages 746–760. Springer, 2012.

[37] Leslie N Smith and Nicholay Topin. Super-convergence: Very fast training of neural networks using large learning rates. In *Artificial intelligence and machine learning for multi-domain operations applications*, pages 369–386. SPIE, 2019.

[38] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019.

[39] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 402–419. Springer, 2020.

[40] Yi-Min Tsai, Yu-Lin Chang, and Liang-Gee Chen. Block-based vanishing line and vanishing point detection for 3d scene reconstruction. In *2006 international symposium on intelligent signal processing and communications*, pages 586–589. IEEE, 2005.

[41] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[42] Kun Wang, Zhenyu Zhang, Zhiqiang Yan, Xiang Li, Baobei Xu, Jun Li, and Jian Yang. Regularizing nighttime weirdness: Efficient self-supervised monocular depth estimation in the dark. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 16055–16064, 2021.

[43] Kun Wang, Zhiqiang Yan, Huang Tian, Zhenyu Zhang, Xiang Li, Jun Li, and Jian Yang. Altnerf: Learning robust neural radiance field via alternating depth-pose optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 5508–5516, 2024.

[44] Lijun Wang, Jianming Zhang, Oliver Wang, Zhe Lin, and Huchuan Lu. Sdc-depth: Semantic divide-and-conquer network for monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[45] Yan Wang, Wei-Lun Chao, Divyansh Garg, Bharath Hariharan, Mark Campbell, and Kilian Q Weinberger. Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8445–8453, 2019.

[46] Zhenda Xie, Zigang Geng, Jingcheng Hu, Zheng Zhang, Han Hu, and Yue Cao. Revealing the dark secrets of masked image modeling. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14475–14485, 2023.

[47] Dan Xu, Wei Wang, Hao Tang, Hong Liu, Nicu Sebe, and Elisa Ricci. Structured attention guided convolutional neural fields for monocular depth estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3917–3925, 2018.

13

[48] Zhiqiang Yan, Xiang Li, Kun Wang, Zhenyu Zhang, Jun Li, and Jian Yang. Multi-modal masked pre-training for monocular panoramic depth completion. In *European Conference on Computer Vision*, pages 378–395. Springer, 2022.

[49] Zhiqiang Yan, Kun Wang, Xiang Li, Zhenyu Zhang, Jun Li, and Jian Yang. Rignet: Repetitive image guided network for depth completion. In *Computer Vision – ECCV 2022*, pages 214–230, Cham, 2022. Springer Nature Switzerland.

[50] Zhiqiang Yan, Xiang Li, Kun Wang, Shuo Chen, Jun Li, and Jian Yang. Distortion and uncertainty aware loss for panoramic depth completion. In *International Conference on Machine Learning*, pages 39099–39109. PMLR, 2023.

[51] Zhiqiang Yan, Kun Wang, Xiang Li, Zhenyu Zhang, Jun Li, and Jian Yang. Desnet: Decomposed scale-consistent network for unsupervised depth completion. In *Proceedings of the AAAI conference on artificial intelligence*, pages 3109–3117, 2023.

[52] Zhiqiang Yan, Yuankai Lin, Kun Wang, Yupeng Zheng, Yufei Wang, Zhenyu Zhang, Jun Li, and Jian Yang. Tri-perspective view decomposition for geometry-aware depth completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

[53] Zhiqiang Yan, Yupeng Zheng, Deng-Ping Fan, Xiang Li, Jun Li, and Jian Yang. Learnable differencing center for nighttime depth perception. *Visual Intelligence*, 2(1):15, 2024.

[54] Guanglei Yang, Hao Tang, Mingli Ding, Nicu Sebe, and Elisa Ricci. Transformer-based attention networks for continuous pixel-wise prediction. In *Proceedings of the IEEE/CVF International Conference on Computer vision*, pages 16269–16279, 2021.

[55] 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*, pages 5684–5693, 2019.

[56] Wei Yin, Jianming Zhang, Oliver Wang, Simon Niklaus, Long Mai, Simon Chen, and Chunhua Shen. Learning to recover 3d scene shape from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 204–213, 2021.

[57] Zhu Yu, Zehua Sheng, Zili Zhou, Lun Luo, Si-Yuan Cao, Hong Gu, Huaqi Zhang, and Hui-Liang Shen. Aggregating feature point cloud for depth completion. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8732–8743, 2023.

[58] 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*, pages 3916–3925, 2022.

[59] Ruo Zhang, Ping-Sing Tsai, James Edwin Cryer, and Mubarak Shah. Shape-from-shading: a survey. *IEEE transactions on pattern analysis and machine intelligence*, 21(8):690–706, 1999.

[60] Zhenyu Zhang, Zhen Cui, Chunyan Xu, Yan Yan, Nicu Sebe, and Jian Yang. Pattern-affinitive propagation across depth, surface normal and semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4106–4115, 2019.

[61] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017.

14

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly state our contributions and the claims made well match the experimental results.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We have discussed the limitations of our work in Sec. 5.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

15

Justification: Our paper does not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We have elaborated our architecture design in the Method section, and have described the implementation details in the Experiment section.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

16

Answer: [Yes]

Justification: The datasets used in our work are all publicly available, and we have properly cited these datasets. The code and related materials of our work are publicly available at `https://github.com/w2kun/DCDepth`.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: We have provided the experimental details in the Experiment section.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: The datasets used in our experiments are large, and it is very costly to train our model for many times. In summary, we don't report the error bar.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)

17

- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: We have reported the information of compute resources used in our experiments.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

Answer: [Yes]

Justification: Our research conforms with the NeurIPS Code of Ethics.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: We have discussed the broader impacts of our work in Sec. 5.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

18

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: Our work poses no such risks.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All datasets used in our work are properly cited.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

19

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

   Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

   Answer: [NA]

   Justification: Our paper does not release new assets.

   Guidelines:

   - The answer NA means that the paper does not release new assets.
   - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
   - The paper should discuss whether and how consent was obtained from people whose asset is used.
   - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

   Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

   Answer: [NA]

   Justification: Our paper does not involve crowdsourcing nor research with human subjects.

   Guidelines:

   - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
   - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
   - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

   Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

   Answer: [NA]

   Justification: Our paper does not involve crowdsourcing nor research with human subjects.

   Guidelines:

   - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
   - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
   - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
   - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

20