PatchFusion: An End-to-End Tile-Based Framework for High-Resolution Monocular Metric Depth Estimation

Zhenyu Li, Shariq Farooq Bhat, Peter Wonka
King Abdullah University of Science and Technology (KAUST)
https://zhyever.github.io/patchfusion/
zhenyu.li.1@kaust.edu.sa

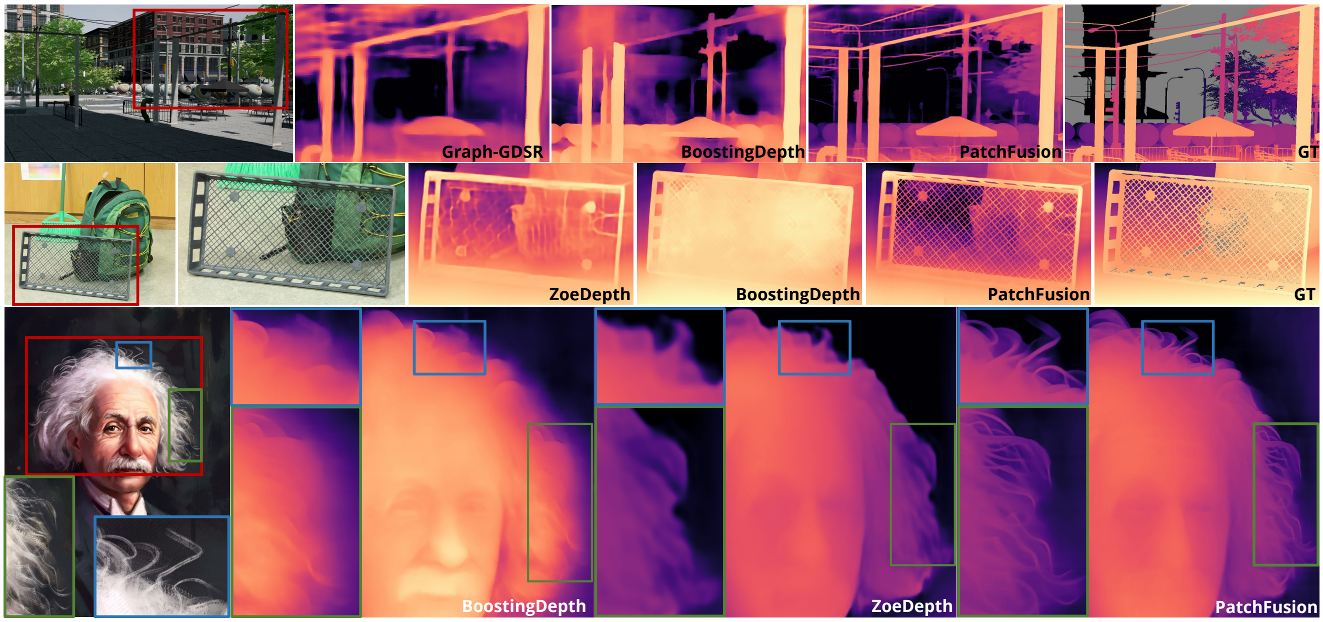


Figure 1. **High-Resolution Depth Estimation.** Our tile-based monocular metric depth estimation model processes high-resolution images and delivers high-quality depth estimations with intricate details on test images corresponding to the synthetic training dataset UnrealStereo4K [37] as well as for zero-shot generalization to other types of real images. **Top:** In-domain sample from UnrealStereo4K. **Middle:** Out-of-domain sample from Middlebury 2014 [35]. **Bottom:** Out-of-domain sample from the internet.

Abstract

Single image depth estimation is a foundational task in computer vision and generative modeling. However, prevailing depth estimation models grapple with accommodating the increasing resolutions commonplace in today's consumer cameras and devices. Existing high-resolution strategies show promise, but they often face limitations, ranging from error propagation to the loss of high-frequency details. We present ***PatchFusion****, a novel tile-based framework with three key components to improve the current state of the art:* ***(1)*** *A patch-wise fusion network that fuses a globally-consistent coarse prediction with finer, inconsistent tiled predictions via high-level feature guidance,* ***(2)*** *A Global-to-Local (G2L) module that adds vital context to the fusion network, discarding the need for patch selection heuristics, and* ***(3)*** *A Consistency-Aware Training (CAT) and Inference (CAI) approach, emphasizing patch overlap consistency and thereby eradicating the necessity for post-processing. Experiments on UnrealStereo4K, MVS-Synth, and Middlebury 2014 demonstrate that our framework can generate high-resolution depth maps with intricate details. PatchFusion is independent of the base model for depth estimation. Notably, our framework built on top of SOTA ZoeDepth brings improvements for a total of 17.3% and 29.4% in terms of the root mean squared error (RMSE) on UnrealStereo4K and MVS-Synth, respectively.*

1. Introduction

This paper addresses the challenge of metric single image depth estimation for high-resolution inputs. Single image depth estimation remains a cornerstone in computer vision and generative modeling [3, 11, 22, 42]. Yet, most state-of-the-art (SOTA) depth estimation architectures are bottlenecked by the resolution capabilities of their backbone. For instance, ZoeDepth [3] processes an input resolution of 384×512, VPD [43] manages 480×480, and AiT [29] is designed for 384×512. These figures pale in comparison to the resolutions offered by modern consumer cameras, such as the 45 Megapixel Canon EOS R5, the widely available 8K televisions, and even mobile devices like the iPhone 15, which boasts a 12MP Ultra Wide lens.

Several methods have attempted to bridge this resolution gap: **(1) Guided Depth Super-Resolution (GDSR)** techniques [20, 26, 44, 45] aim to refine high-resolution depth maps from their low-resolution counterparts using high-resolution color images as reference. **(2) Implicit Function** approaches, such as SMD-Net [37], leverage algorithms like [5, 28] to estimate disparities continuously across image locations, essentially performing on-the-fly super-resolution. However, due to the low-resolution nature of many models, these techniques have their limitations: GDSR can propagate errors, and the implicit function still strips away crucial high-frequency details during input downsampling. Lastly, the **(3) Tile-Based Method** proposed in BoostingDepth [27], emphasizes *relative* depth estimation by processing image patches independently before merging them to form a unified depth map.

Our approach refines the concept of tile-based depth estimation. While the BoostingDepth already has promising results, we identified some shortcomings to improve. First, we discover that BoostingDepth suffers from scale inconsistencies, especially when transposed to *metric* depth estimation. These inconsistencies then mandate rigorous post-process corrections, such as scale optimization and Gaussian blending. Second, the fusion network in BoostingDepth often stumbles due to the lack of guidance and its inability to grasp a more holistic view of the input image, leading to local optima and compelling the use of complex heuristic patch selections.

In light of these challenges, we introduce **PatchFusion** that stands on three pillars: **Firstly**, we augment the fusion network with high-level feature guidance, streamlining its training. **Secondly**, our proposed Global-to-Local (G2L) module empowers the fusion network to stay context-aware, eliminating complex patch selection heuristics. **Thirdly**, our Consistency-Aware Training (CAT) and Inference (CAI) strategy places a special emphasis on patch overlap consistency, facilitating consistency-aware training and inference. As a result, we achieve an end-to-end tile-based framework without any necessity for pre-processing such as patch selection, and post-processing like scale optimization and Gaussian blending. In Fig. 1 we show selected examples to illustrate how PatchFusion is able to significantly improve the results compared to the previous SOTA. In summary, our key contributions include:

- The introduction of a novel tile-based network architecture and training strategy for metric monocular depth estimation called PatchFusion. PatchFusion is adept at handling high-resolution images and is the first tile-based metric depth estimations approach that can be trained in an end-to-end manner without the need for additional post-processing or pre-processing steps.
- We conducted exhaustive empirical validations using the UnrealStereo4K [37], MVS-Synth [19], and Middleburry 2014 [35] datasets. Our framework further improves current SOTA by 17.3% and 29.4% on UnrealStereo4K and MVS-Synth, respectively.

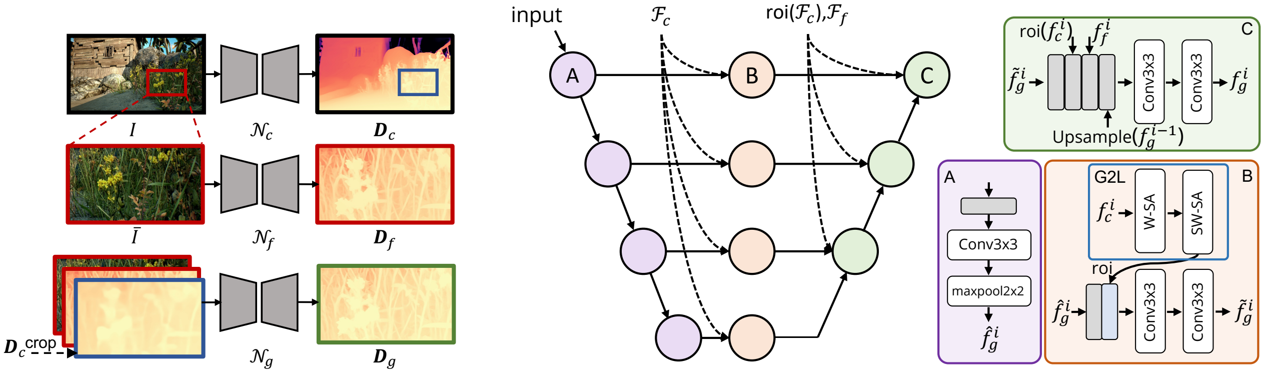
2. Related Work

2.1. Monocular Depth Estimation

Tremendous progress in monocular depth estimation has been achieved by publicly available large-scale datasets [6, 7, 14, 36], network design [2, 11, 22, 23, 38, 41], loss supervision [21, 24, 40], refined problem formulations [2, 9, 13, 40], and training strategies [12, 15, 30]. Recently, the best-performing frameworks have been built on the transformer architecture [1, 10, 25]. While current SOTA frameworks demonstrate exceptional performance, they still use low-resolution images as input. For example, the SOTA ZoeDepth [3] uses BEIT$_{384}$-L [1] with 307M parameters and only infers 384×512 (about 0.2 megapixel) images. This stands in stark contrast to the advancements in modern imaging devices that capture images at increasing resolutions, and the growing demand among users for high-resolution depth estimation. In this work, we aim to utilize these large-scale models on high-resolution inputs and achieve high-resolution depth estimation.

2.2. High-Resolution Depth

The pursuit of high-resolution depth estimation has typically converged on three prevalent strategies: **(1) Guided Depth Super-Resolution (GDSR)**: This approach employs methods [20, 26, 44, 45] to reconstruct high-resolution depth maps using low-resolution depth observations. These reconstructions are facilitated by paired high-resolution color images. A notable limitation of GDSR is its reliance on training samples derived from the direct downsampling of high-resolution depth maps. When applied to monocular depth estimation—where low-resolution depth maps stem from models limited by input resolution constraints—this approach can introduce cascading errors during the super-resolution process. **(2) Implicit Function**: Melding the



(a) Overall Pipeline

(b) Guided Fusion Network

Figure 2. **Framework Illustration.** **(a) Overall Pipeline:** The framework consists of three phases, including a coarse network $\mathcal{N}_c$ providing globally consistent yet coarse depth estimation $\mathbf{D}_c$, a fine network $\mathcal{N}_f$ presenting finer details for an input tile (but all tiles together would lack consistency), and a guided fusion network $\mathcal{N}_g$ combining the best of both worlds. **(b) Architecture of Guided Fusion Network:** The lightweight network includes (A) successive encoder layers, (B) skip-connection modules, and (C) upsampling layers.

implicit function [5, 28] with stereo mixture density has enabled architectures like SMD-Net [37] to deliver precise disparity estimations across continuous image locations, thereby addressing the stereo super-resolution challenge. Nevertheless, this method hasn't fully confronted the constraints posed by image resolution. Models are still vulnerable to losing intricate high-frequency details during the input downsampling phase. **(3) Tile-Based Method**: This paper aligns itself with the tile-based strategy, pioneered in works such as BoostingDepth [27] and [33]. The underlying principle here is to segment depth estimation into patches. Subsequently, these patches are merged to construct a holistic image depth estimation. A chief advantage of this method is its ability to sidestep the severe downsampling often mandated by the restrictions on input resolution.

2.3. Depth Map Blending

Given coarse whole-image and fine patch-wise depth maps, blending them together is a crucial part of tile-based methods. This process aims to keep the correct global scale in coarse maps while maintaining the fine details in the tile maps. BoostingDepth [27] applies a linear polynomial, whereas [33] utilizes a deformable depth field proposed in [18] to achieve the blending. The former optimizes one patch-wise scale and shift in the least squares manner and the latter calculates pixel-wise deformation. Unlike previous approaches, the output of our method can be seamlessly stitched without the necessity for any post-optimization, resulting in an end-to-end framework for high-resolution depth estimation.

3. Method

In this section, we present the overall framework in Sec. 3.1, Consistency-Aware Training and Inference in Sec 3.2, and implementation details in Sec. 3.3.

3.1. Overall Framework

Our primary objective is to harness the capabilities of a pre-trained base depth estimation model trained on low resolution and employ it for high-resolution depth estimation, targeting high resolutions, e.g. 4K. Unfortunately, a straightforward scaling of current base models to high resolutions such as 4K far exceeds the memory and compute capabilities of current hardware. Therefore, we adopt a patch-wise approach, breaking down the high-resolution depth estimation task into three distinct steps (see Fig. 2a): **(i)** Global Scale-Aware Estimation, **(ii)** Local Fine-Depth Estimation, and **(iii)** Fusion. We train a dedicated network for each step as described below.

(i) Global Scale-Aware Estimation: In the initial step, we predict a coarse depth map by first downsampling the input image to the native resolution of the depth model. This downsampling reduces the memory requirements and computational load while providing an initial estimation of the depth, $\mathbf{D}_c$. For this step, we fine-tune our base depth model on the downsampled data, resulting in the coarse network $\mathcal{N}_c$. Due to downsampling, the output from this step is coarse in nature and fine, high-frequency details are lost at the cost of global consistency.

(ii) Patch-Wise Depth Prediction: In this step, we divide the input images into smaller manageable patches and feed the cropped patches as input to our base model. We use a fixed patch size that is equal to or similar to the native resolution of the base depth model. This results in a fine prediction $\mathbf{D}_f$ containing rich details, particularly at boundaries and intricate structures. Nevertheless, this detailed depth map, being confined to only a segment of the original scene, remains oblivious to the global context, mak-

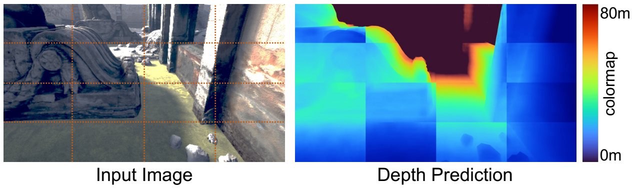


Figure 3. **Illustration of Inconsistent Depth Prediction.** Patch-wise depth prediction by itself suffers from the lack of global information, leading to inconsistent depth predictions especially visible at the patch boundaries.

ing its scale potentially inconsistent with the actual scene. This could result in a scale-shifted prediction and fluctuations across patches leading to obvious patch artifacts as shown in Fig. 3.

(iii) Fusion and Guided Fusion Network: To resolve the inconsistency issues with patch-wise prediction, our goal in this step is to transfer the global information from the coarse depth obtained in **(i)** to the patch-wise predictions without compromising their higher details. While the transfer could be implemented by simply learning a pix2pix U-Net [27, 34], our key idea is to exploit the multi-scale features from $\mathcal{N}_c$ and $\mathcal{N}_f$. We use two main components to achieve this transfer - the Global-to-Local Module (G2L) and the Guided Fusion Network $\mathcal{N}_g$.

Global-to-Local (G2L) Module: Our empirical evaluations (in Tab. 2) indicate that directly adopting the global guidance feature $\mathcal{F}_c = \{f_c^i\}_{i=1}^L$ from $\mathcal{N}_c$ (where L is the number of layers) still suffers from the scale-shift issue. Even though $\mathcal{F}_c$ is derived from the entirety of the image, the necessary information needed for accurate scale inference during fusion is lost post the cropping operation. Addressing this, we present our G2L module designed to retain global context.

While the key insight of G2L is to apply the global-wise self attention for each-level feature in $\mathcal{F}_c$ to ensemble crucial information for patch-wise scale-consistent prediction, we adopt the Swin Transformer Layer (STL) [25] to preserve the global context while simultaneously alleviating GPU memory concerns. The main ideas are the local attention and the shifted window mechanism. Given each feature map f_c^i, STL subdivides it into localized windows for self-attention (W-SA), which is then followed by shifted-window attention for inter-window interactions (SW-SA). The operation can be formulated as:

$$f_{g2l}^i = \texttt{G2L}(f_c^i) = \texttt{SW-SA}(\texttt{W-SA}(f_c^i)), \qquad (1)$$

where the superscript i iterates through the multi-layer features. The output set $\mathcal{F}_{g2l} = \{f_{g2l}^i\}_{i=1}^L$ is then sent to the Guided Fusion Network $\mathcal{N}_g$ for further fusion.

Guided Fusion Network: The guided fusion network follows the U-Net [34] design as shown in Fig. 2b. The input comprises a concatenated ensemble of the cropped original image I, the corresponding cropped coarse depth estimations $\mathbf{D}_c$ from $\mathcal{N}_c$, and fine depth estimations $\mathbf{D}_f$ from $\mathcal{N}_f$. The key idea of the design is to only use image and depth values in the encoder and to delay the injection of network features to the skip connections and decoder layers.

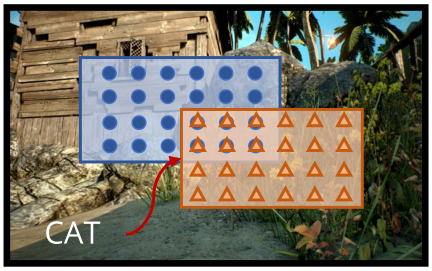


Figure 4. **Consistency-Aware Training Illustration.** We apply our consistency loss on the overlap area of intermediate features and depth predictions from two patches.

We use a lightweight encoder consisting of successive convolutional and max-pooling layers to extract multi-level features. During the skip-connection, we inject the scale-aware feature $\mathcal{F}_{g2l}$ with a fusion block ($\texttt{FB}$) consisting of two 3 × 3 convolutional layers with $\texttt{ReLU}$ activation functions as

$$\tilde{f}_g^i = \texttt{FB}(\texttt{roi}(f_{g2l}^i), \hat{f}_g^i), \qquad (2)$$

where we apply the $\texttt{roi}$ [17] operation to fetch features of the corresponding cropped area. $\hat{f}_g^i$ denotes the initial output feature from the i-th encoder layer. As for the decoder, we again harness the guidance features $\mathcal{F}_c = \{f_c^i\}_{i=1}^L$ and $\mathcal{F}_f = \{f_f^i\}_{i=1}^L$ from $\mathcal{N}_c$ and $\mathcal{N}_f$, respectively, integrating them into the decoder of our fusion model. The operation can be formulated as:

$$f_g^i = \texttt{FB}(\tilde{f}_g^i, \texttt{roi}(f_c^i), f_f^i, \texttt{Upsample}(f_g^{i-1})) \qquad (3)$$

where $\tilde{f}_g^i$ is obtained from Eq. 2. The $\texttt{Upsample}$ function 2× rescales the features from the previous level in the decoder. The output features $\mathcal{F}_g = \{f_g^i\}_{i=1}^L$ are then sent to a depth head [3] for final depth estimation.

3.2. Consistency-Aware Training and Inference

The effectiveness of the fusion network hinges on not just the accuracy of predictions, but also their consistency across patch boundaries. While our Guided Fusion Network makes scale-aware predictions, boundary inconsistencies still exist. Recognizing this gap, we introduce Consistency-Aware Training (CAT) and Inference (CAI) to ensure patch-wise depth prediction consistency.

Training: Our methodology, see Fig. 4, is based on the intuitive idea that overlapping regions between cropped patches from the same image should ideally produce consistent feature representations and depth predictions. We begin by cropping an image patch, denoted as I_1. By shifting the cropping window, we obtain another cropped patch I_2, such that there exists an overlap region Ω. Both these patches, I_1 and I_2, are independently processed through our framework, yielding image features $\mathcal{F}_g^1$ and $\mathcal{F}_g^2$, and depth predictions $\mathbf{D}_1$ and $\mathbf{D}_2$, respectively. To enforce consistency, we impose an L_2 loss on the overlapping regions of both the extracted image features and the depth predictions. This loss penalizes discrepancies in the feature representations and depth predictions of the overlapping regions from the two cropped patches. The consistency-aware loss can be mathematically expressed as:

$$\mathcal{L}_c = ||\mathcal{F}_g^1 - \mathcal{F}_g^2||_2 + \mu_1||\mathbf{D}_1 - \mathbf{D}_2||_2,\ x \in \Omega, \quad (4)$$

where c is short for consistency, μ_1 is a hyperparameter empirically set to 0.1 to balance the loss. While the idea of constraining the depth values is quite intuitive, the good results mainly stem from constraining the features.

Inference: One of the distinct advantages of our method, especially when compared with BoostingDepth [27], is its freedom from heuristic patch selection and post-processing strategies during inference. In our standard inference pipeline, we slice the image into P = 16 non-overlapping patches, spanning its entirety. These patches are processed, and the depth maps are seamlessly stitched together. This standard pipeline is called PatchFusion$_{\mathrm{P=16}}$ in Tab. 1.

To unlock the full power of the network, we complement consistency-aware inference in the following manner. Our model is further amplified with the inclusion of an extra 33 shifted, tidily arranged patches, as illustrated in the supplementary material, as PatchFusion$_{\mathrm{P=49}}$. An additional improvement is to use extra randomly sampled patches, resulting in PatchFusion$_{\mathrm{R}}$. Unless otherwise specified, we use $R = 128$ random patches in our experiments.

During the processing of patches, the updated depth $\mathbf{D}_g$ is concatenated with the cropped image I and coarse depth map $\mathbf{D}_c$, supplanting the $\mathbf{D}_f$, as the input to our guided fusion network. This dynamic updating, coupled with a running mean, engenders a local ensemble[1] approach, incrementally refining the depth estimations on the fly.

3.3. Implementation Details

Training: Both networks, $\mathcal{N}_c$ and $\mathcal{N}_f$, are trained utilizing the scale-invariant loss $\mathcal{L}_{si}$ [3, 11, 22]. We use weights from pretraining on the NYU-v2 dataset [36] as our initialization. $\mathcal{N}_c$ and $\mathcal{N}_f$ are then fine-tuned on the target dataset for 16 epochs. The training of the guided fusion network involves a combination of the scale-invariant loss $\mathcal{L}_{si}$ and our specially designed consistency loss $\mathcal{L}_c$ as

$$\mathcal{L} = \mathcal{L}_{si} + \mu_2\mathcal{L}_c, \quad (5)$$

where μ_2 is a hyperparameter empirically set to 0.1 to balance the loss. The fusion network $\mathcal{N}_g$ is trained for 12 epochs. Data augmentation for $\mathcal{N}_f$ and $\mathcal{N}_g$ includes random cropping. Beyond this, we use the default augmentation strategies adopted in the baseline depth model.

4. Experimental Results

4.1. Datasets and Metrics

UnrealStereo4K: The UnrealStereo4K dataset [37] offers synthetic stereo images in 4K resolution (2160×3840), complete with accurate, pixel-wise ground truth. Since the dataset has some incorrectly labeled images, we use the Structural Similarity Index (SSIM) [39] for quality assurance, comparing the original and reconstructed left images with the given disparity maps. Entries with SSIM below 0.7 are excluded (we filter 131 out of 7860 images total). Utilizing the provided camera parameters, we convert the disparity maps to metric depth maps. Our experiments follow the prescribed dataset splits in [37], and we select a patch size of 540×960 by default.

MVS-Synth: MVS-Synth [19] is a synthetic dataset designed for training Multi-View Stereo (MVS) algorithms. It contains 120 unique sequences, each with 100 frames depicting urban scenes from the virtual environment of Grand Theft Auto V. Each frame provides a 2K (1080×1920) RGB image, along with a corresponding ground truth depth map and camera parameters. We divide the dataset into 11,160 training samples and 240 validation pairs, cropping images into 270×480 patches for inputs to $\mathcal{N}_f$ and $\mathcal{N}_g$.

Middlebury 2014: The Middlebury 2014 dataset [35] comprises a set of high-resolution stereo images (almost 4K), showcasing indoor scenes in controlled lighting settings. It includes 23 images paired with corresponding ground-truth disparity maps. Direct training of a monocular metric depth estimation model on this dataset is not advised due to the risk of overfitting. We use the dataset to test our zero-shot generalization capability, particularly from synthetic to real-world scenarios.

Metrics: We use the standard metric depth evaluation metrics proposed in [3, 11, 31] and present details in the supplementary material. Furthermore, we introduce two additional metrics to specifically evaluate the precision at object boundaries and the consistency across patches: (1) Soft Edge Error (SEE): As per the recommendations in [4, 37], SEE assesses the fidelity of boundary estimation. It measures the discrepancy between predicted disparity and the ground truth within a 3×3 local patch around object edges, penalizing any over-smoothing tendencies and underlining

[1] Ensemble is combining several different predictions from different models to make the final prediction. Here, we abuse this term to represent combining different predictions from different patches but using the same model

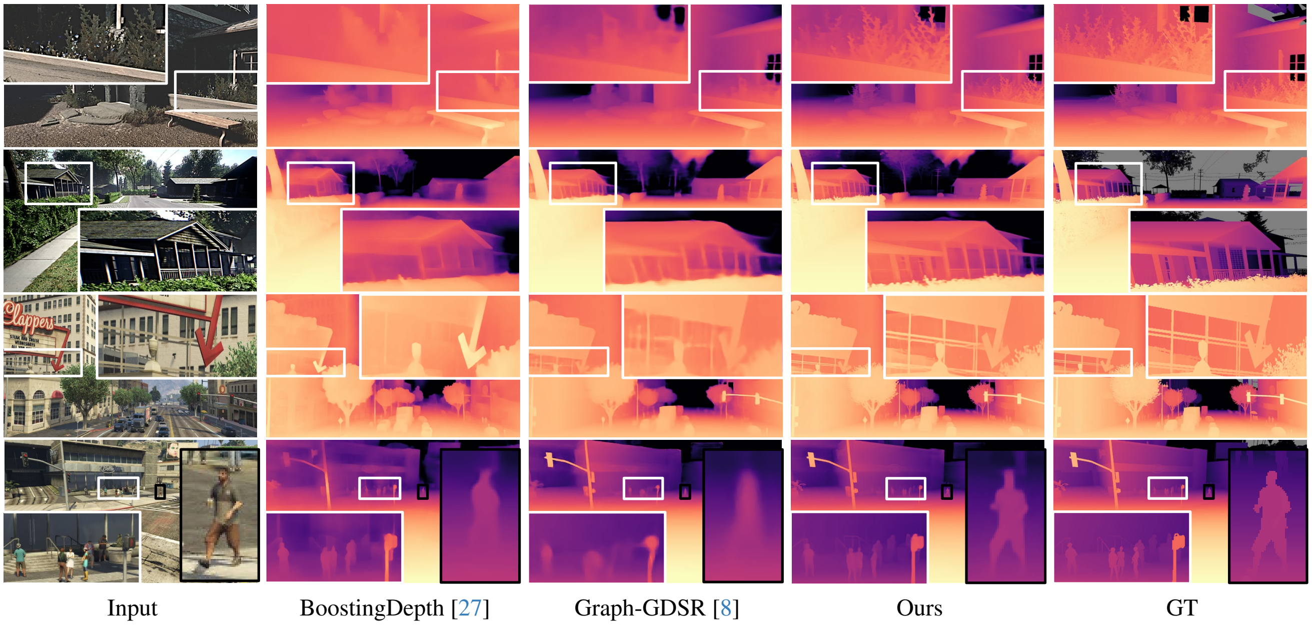


Figure 5. **Qualitative Results on UnrealStereo4K and MVS-Synth.** The qualitative comparisons showcased here are derived from the UnrealStereo4K and MVS-Synth datasets, depicted in the first two and last two rows, respectively. Zoom in to better perceive details near boundaries. Our framework outperforms counterparts [8, 27] with much sharper edges.

the model's ability to capture high-fidelity depth contours. (2) Consistency Error (CE): To ascertain patch coherence, CE is calculated as the mean absolute error among patches designed with half-resolution overlaps. This metric evaluates the uniformity of the depth estimation across different patches, ensuring a seamless transition and consistency in the reconstructed depth map.

4.2. Main Results

Our framework can enhance any metric depth estimation model. For our experiments, we select ZoeDepth [3] as our baseline due to its SOTA performance. We compare our framework against various other methods, including (1) traditional monocular metric depth estimation methods such as ZoeDepth [3] and iDisc [31], (2) the guided depth super-resolution approach Graph-GDSR [8], (3) SMD-Net [37] which utilizes an implicit function head for enhanced depth estimation, and (4) the tile-based technique BoostingDepth [27]. To ensure fair comparisons, we standardize the depth models across these strategies to ZoeDepth or we fine-tune the models using their provided NYU-v2 [36] pretrained versions on both the UnrealStereo4K [37] and MVS-Synth [19] datasets, setting this as our default experimental protocol.

UnrealStereo4K: The performance comparison on the UnrealStereo4K test set is presented on the left of Tab. 1. While all compared methods enhance prediction quality to some degree, our PatchFusion framework significantly surpasses the baseline ZoeDepth model by 17.3% in RMSE and 14.7% in REL. This indicates a considerable performance gain afforded by our approach. Our results also showcase the lowest Soft Edge Error (SEE), evidencing superior boundary delineation. Qualitative comparisons in Fig. 5 further underscore the high-quality depth maps generated by our framework. We capture intricate details, especially in boundary areas. More cases are shown in our supplementary material.

MVS-Synth: The right of Tab. 1 provides the quantitative comparison results for experiments on the MVS-Synth dataset [19]. Our framework outperforms all alternative approaches in every evaluated metric. Notably, our technique boosts the RMS score by approximately 29.4% and reduces the scale-invariant logarithmic error by 30.8%. We present qualitative results in Fig. 5 and the supplementary material.

Middlebury 2014: We first evaluate the zero-shot transfer capability of our framework. The accompanying supplementary material provides both quantitative and qualitative comparisons to illustrate the efficacy of our approach. Building on this foundation, we explored our framework's application to the field of text-to-image generation. We substituted the conventional depth estimation model [32] employed by ControlNet [42] with our PatchFusion. To ensure compatibility with the depth range required by ControlNet, we normalized the metric depth output of PatchFusion to the interval $[0, 256]$ and then applied the transformation $256 - \hat{d}$ to invert it. The results are illustrated in

Method	UnrealStereo4K					MVS-Synth				
	δ_1 (%)↑	**REL**↓	**RMS**↓	**SiLog**↓	**SEE**↓	δ_1 (%)↑	**REL**↓	**RMS**↓	**SiLog**↓	**SEE**↓
iDisc [31]	96.940	0.0534	1.4035	8.5022	1.0697	93.010	0.0866	1.4386	15.4157	1.5624
BoostingDepth* [27]	75.483	0.1890	4.7310	56.3251	3.3204	71.393	0.2731	4.6859	85.8841	4.1082
BoostingDepth [27]	98.104	0.0437	1.1233	6.6623	0.9390	95.409	0.0694	0.9535	11.5144	1.2694
Graph-GDSR* [8]	97.757	0.0454	1.3012	7.6316	0.8734	94.075	0.0760	1.2735	14.8825	1.2386
Graph-GDSR [8]	97.932	0.0438	1.2642	7.4691	0.8718	94.195	0.0748	1.2435	14.0723	1.2106
SMD-Net [37]	97.774	0.0439	1.2817	7.3888	0.8828	93.842	0.0776	1.2563	14.1074	1.2747
ZoeDepth$_{\text{COARSE}}$ [3]	97.717	0.0455	1.2887	9.1227	0.9144	93.978	0.0769	1.2676	14.1236	1.3036
ZoeDepth$_{\text{FINE}}$ [3]	97.027	0.0627	1.2058	7.4483	0.9546	95.113	0.0715	0.9454	11.3844	1.1142
ZoeDepth+PF$_{P=16}$	98.419	0.0399	1.0878	6.2122	**0.8382**	95.991	0.0613	0.9213	10.1511	<u>1.0759</u>
ZoeDepth+PF$_{P=49}$	<u>98.450</u>	<u>0.0392</u>	<u>1.0747</u>	<u>6.1311</u>	<u>0.8462</u>	<u>96.069</u>	<u>0.0599</u>	<u>0.9050</u>	<u>9.9524</u>	**1.0700**
ZoeDepth+PF$_{R=128}$	**98.469**	**0.0388**	**1.0655**	**6.0846**	0.8488	**96.172**	**0.0589**	**0.8944**	**9.7696**	1.0833

Table 1. **Quantitative comparison on UnrealStereo4K and MVS-Synth.** Best results are in **bold**, second best are <u>underlined</u>. * indicates out of bbox inference without training on the target datasets. PF is short for PatchFusion.

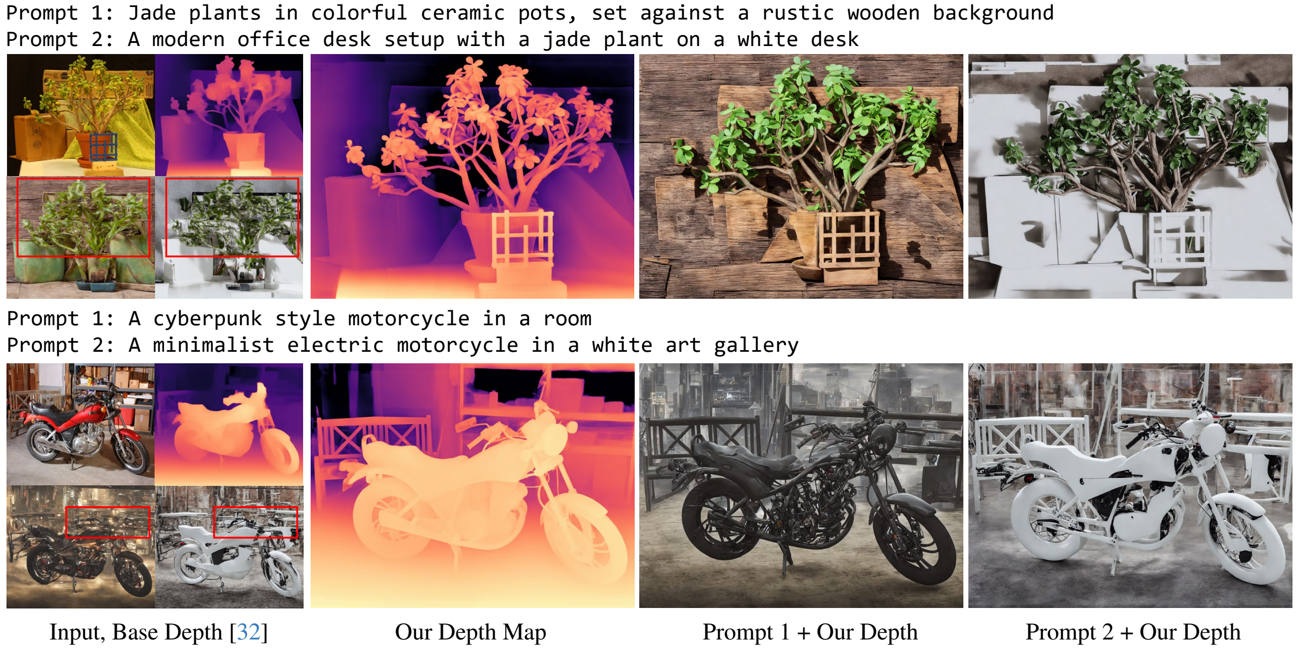

Prompt 1: Jade plants in colorful ceramic pots, set against a rustic wooden background
Prompt 2: A modern office desk setup with a jade plant on a white desk
Prompt 1: A cyberpunk style motorcycle in a room
Prompt 2: A minimalist electric motorcycle in a white art gallery
Input, Base Depth [32]
Our Depth Map
Prompt 1 + Our Depth
Prompt 2 + Our Depth

Figure 6. **Qualitative Results of Depth-Guided Text-to-Image Generation.** We show two cases each with two different text prompts. Given the image, we use the default depth estimation model [32] in ControlNet [42] and our PatchFusion to predict depth maps, respectively. Along with the same prompt, the depth maps are processed by the text-to-image model in ControlNet to generate images. The depth resolution is aligned by rescaling. We show some images in a smaller size to save space. Zoom in for details like the leaves and handlebar marked by red boxes. The images are from Middlebury 2014 [35].

Fig. 6. Armed with the high-fidelity depth maps generated by PatchFusion, which capture intricate details, the text-to-image generation model yields coherent structures, textures, and details of significantly higher quality than the baseline.

4.3. Ablation Studies and Discussion

In this section, we ablate and discuss the contributions of individual components within our framework. Unless stated otherwise, we utilize the UnrealStereo4K dataset and the PatchFusion variant with $P = 16$ patches for clarity and ease of comparison. Inference time benchmarks are performed on a single NVIDIA A100 GPU.

Comparison with other Blending Strategies: The primary role of the fusion network in our framework is to act as an adaptive blending module, integrating the fine detail from $\mathbf{D}_f$ with the scale information retained in $\mathbf{D}_c$. We benchmark our network against other blending strategies prevalent in various tile-based depth estimation methods [18, 27, 33]. As illustrated in Tab. 2 (①, ②, ③), our approach yields superior results in terms of estimation accuracy and satisfactory computational efficiency. We highlight that traditional blending strategies often grapple with

	G	G2L	CAT-P	CAT-F	Method	**REL**↓	**SEE**↓	**CE**↓	**T**(s) ↓
					Coarse Baseline	0.0455	0.9144	-	0.080
					Fine Baseline	0.0627	0.9546	0.2546	1.132
①					Patch Global Scale&Shift opt.	0.0544	1.0060	0.3703	2.256
②					Pixel Deformation opt.	0.0437	0.9016	0.2213	9.729
③					Poisson Image Editing opt.	0.0467	1.0313	0.2499	31.875
④						0.0450	0.8932	0.2155	1.556
⑤	✓					0.0423	0.8588	0.2318	1.642
⑥	✓		✓			0.0419	0.8415	0.1700	
⑦	✓	✓				0.0414	0.8473	0.1714	2.782
⑧	✓	✓	✓			0.0411	0.8624	0.1215	
⑨	✓	✓		✓		0.0403	0.8451	0.1624	
⑩	✓	✓	✓	✓		0.0399	**0.8382**	0.1441	
					PatchFusion$_{P=49}$ (w/o CAI)	0.0398	0.8397	0.1441	7.818
★					PatchFusion$_{P=49}$ (full method)	**0.0392**	0.8462	**0.0464**	

Table 2. Ablation study on UnrealStereo4K dataset. G denotes the feature guidance. CAT-P and CAT-F denote adopting consistency-aware training on predictions and features, respectively. Best results are in **bold**, second best are underlined. Time: average inference time for one image.

the challenge of specifying an optimal handcrafted target for optimization, which can result in suboptimal outcomes. Our network circumvents this issue by learning to blend the input patches effectively through a data-driven process. This is evidenced by the substantial gains in performance metrics, with an 8.7% improvement in REL and a 34.8% reduction in CE during inference.

Guided Fusion Network with G2L Module: Our initial focus is to evaluate the impact of the Guided Fusion Network equipped with the G2L (Global to Local) Module. We incrementally introduce the guidance feature and the G2L module to a baseline encoder-decoder structure, similar to the MergeNet in BoostingDepth [27], and observe the changes in performance. As indicated in Tab. 2, the standalone model without the guidance feature and the G2L module (④) yields the least impressive results. With RMSE and SEE metrics closely aligning with the coarse baseline, and only a marginal improvement in the consistency error, it becomes evident that this variant struggles to integrate the finer details from $\mathbf{D}_f$ into the coarse depth map $\mathbf{D}_c$. This underlines the crucial role our framework plays in surpassing the capabilities of BoostingDepth [27].

Introducing the guidance feature (⑤) marks an uptick in performance with improvements of 6.0% in REL and 3.8% in SEE, confirming the feature's role in enhancing training and the assimilation of fine details. The integration of the G2L module (⑦) leads to a 26.0% reduction in consistency error, achieved without imposing explicit constraints. The concomitant reduction in 2.1% REL underscores the G2L module's effectiveness in harnessing global information, thus improving both accuracy and consistency.

Consistency-Aware Training and Inference: Integrating the consistency-aware training (CAT) loss, results in enhanced REL and CE metrics for models, both with and without the G2L module (⑥ and ⑧, respectively). Notably, the introduction of the G2L module and consistency loss leads to a marginal reduction in SEE. This suggests that the model may be excessively penalized, potentially neglecting some fine details in the process. When the consistency loss is applied to the intermediate features (⑨), we observe a more pronounced improvement in REL, while SEE is preserved, indicating a balanced detail capture. Nevertheless, this approach yields a diminished CE, suggesting a slight trade-off. Upon concurrently enforcing consistency constraints on both predictions and intermediate features, we note a further refinement in both REL and SEE metrics, along with an acceptable level of CE (⑩). This finding underscores the balanced improvements in standard metrics and patch-wise consistency, which is pivotal for a tile-based framework. Collectively, these results affirm the CAT's role in bolstering high-quality depth estimation. Finally, with the use of additional patches and our consistency-aware inference (CAI), we obtain another significant boost in REL and CE (★) due to the local ensemble, but at the expense of SEE and time complexity.

4.4. Limitations and Future Work

One limitation is the computational efficiency of our framework. While incorporating an increasing number of randomly selected patches does improve depth prediction, it also results in a proportional increase in processing time. We therefore suggest an efficient patch selection strategy as an avenue for future work. Another limitation is the lack of high-resolution real-world training data. We can observe this gap in our results from synthetic to real transfer. While the results showcase sharp edges, the scale of the results could be improved. Also, we noticed that UnrealStereo4K [37] does not contain enough images with large homogeneous foreground objects. We believe that the collection of large real-world high-resolution depth datasets will be a great contribution, as depth estimation, in general, seems to be a highly valuable pre-training task [16], and therefore recommend this as future work.

5. Conclusions

We presented **PatchFusion**, an end-to-end tile-based framework tailored to high-resolution monocular metric depth estimation. We introduced a novel tile-based network architecture together with a consistency-aware training and inference strategy. This combination yields a framework that only relies on forward passes through networks and obviates the need for additional pre-processing and post-processing. Our proposed framework decisively improves upon the baseline model for UnrealStereo4K (17.3% in RMSE) and MVS-Synth (29.4% in RMSE), while demonstrating satisfactory performance in zero-shot transfer.

References

[1] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. Beit: Bert pre-training of image transformers. *arXiv preprint arXiv:2106.08254*, 2021. 2

[2] Shariq Farooq Bhat, Ibraheem Alhashim, and Peter Wonka. Adabins: Depth estimation using adaptive bins. In *CVPR*, pages 4009–4018, 2021. 2

[3] Shariq Farooq Bhat, Reiner Birkl, Diana Wofk, Peter Wonka, and Matthias Müller. Zoedepth: Zero-shot transfer by combining relative and metric depth. *arXiv preprint arXiv:2302.12288*, 2023. 2, 4, 5, 6, 7

[4] Chuangrong Chen, Xiaozhi Chen, and Hui Cheng. On the over-smoothing problem of cnn based disparity estimation. In *ICCV*, pages 8997–9005, 2019. 5

[5] Yinbo Chen, Sifei Liu, and Xiaolong Wang. Learning continuous image representation with local implicit image function. In *CVPR*, pages 8628–8638, 2021. 2, 3

[6] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, pages 3213–3223, 2016. 2

[7] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *CVPR*, pages 5828–5839, 2017. 2

[8] Riccardo De Lutio, Alexander Becker, Stefano D'Aronco, Stefania Russo, Jan D Wegner, and Konrad Schindler. Learning graph regularisation for guided super-resolution. In *CVPR*, pages 1979–1988, 2022. 6, 7

[9] Raul Diaz and Amit Marathe. Soft labels for ordinal regression. In *CVPR*, pages 4738–4747, 2019. 2

[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. 2

[11] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. *NeurIPS*, 27, 2014. 2, 5

[12] Rizhao Fan, Matteo Poggi, and Stefano Mattoccia. Contrastive learning for depth prediction. In *CVPRW*, pages 3225–3236, 2023. 2

[13] Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, and Dacheng Tao. Deep ordinal regression network for monocular depth estimation. In *CVPR*, pages 2002–2011, 2018. 2

[14] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *IJRR*, 32 (11):1231–1237, 2013. 2

[15] Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J Brostow. Digging into self-supervised monocular depth estimation. In *ICCV*, pages 3828–3838, 2019. 2

[16] Micah Goldblum, Hossein Souri, Renkun Ni, Manli Shu, Viraj Prabhu, Gowthami Somepalli, Prithvijit Chattopadhyay, Mark Ibrahim, Adrien Bardes, Judy Hoffman, et al. Battle of the backbones: A large-scale comparison of pretrained models across computer vision tasks. *arXiv preprint arXiv:2310.19909*, 2023. 8

[17] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 4

[18] Peter Hedman and Johannes Kopf. Instant 3d photography. *ACM Transactions on Graphics (TOG)*, 37(4):1–12, 2018. 3, 7

[19] Po-Han Huang, Kevin Matzen, Johannes Kopf, Narendra Ahuja, and Jia-Bin Huang. Deepmvs: Learning multi-view stereopsis. In *CVPR*, 2018. 2, 5, 6

[20] Tak-Wai Hui, Chen Change Loy, and Xiaoou Tang. Depth map super-resolution by deep multi-scale guidance. In *ECCV*, pages 353–369. Springer, 2016. 2

[21] Jae-Han Lee and Chang-Su Kim. Multi-loss rebalancing algorithm for monocular depth estimation. In *ECCV*, pages 785–801. Springer, 2020. 2

[22] Zhenyu Li, Xuyang Wang, Xianming Liu, and Junjun Jiang. Binsformer: Revisiting adaptive bins for monocular depth estimation. *arXiv preprint arXiv:2204.00987*, 2022. 2, 5

[23] Zhenyu Li, Zehui Chen, Xianming Liu, and Junjun Jiang. Depthformer: Exploiting long-range correlation and local information for accurate monocular depth estimation. *Machine Intelligence Research*, pages 1–18, 2023. 2

[24] Ce Liu, Suryansh Kumar, Shuhang Gu, Radu Timofte, and Luc Van Gool. Single image depth prediction made better: A multivariate gaussian take. In *CVPR*, pages 17346–17356, 2023. 2

[25] 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 *ICCV*, pages 10012–10022, 2021. 2, 4

[26] Nando Metzger, Rodrigo Caye Daudt, and Konrad Schindler. Guided depth super-resolution by deep anisotropic diffusion. In *CVPR*, pages 18237–18246, 2023. 2

[27] S Mahdi H Miangoleh, Sebastian Dille, Long Mai, Sylvain Paris, and Yagiz Aksoy. Boosting monocular depth estimation models to high-resolution via content-adaptive multi-resolution merging. In *CVPR*, pages 9685–9694, 2021. 2, 3, 4, 5, 6, 7, 8

[28] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 2, 3

[29] Jia Ning, Chen Li, Zheng Zhang, Zigang Geng, Qi Dai, Kun He, and Han Hu. All in tokens: Unifying output space of visual tasks via soft token. *arXiv preprint arXiv:2301.02229*, 2023. 2

[30] Andra Petrovai and Sergiu Nedevschi. Exploiting pseudo labels in a self-supervised learning framework for improved monocular depth estimation. In *CVPR*, pages 1578–1588, 2022. 2

[31] 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. 5, 6, 7

[32] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE TPAMI*, 44(3), 2022. 6, 7

[33] Manuel Rey-Area, Mingze Yuan, and Christian Richardt. 360monodepth: High-resolution 360deg monocular depth estimation. In *CVPR*, pages 3762–3772, 2022. 3, 7

[34] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015. 4

[35] Daniel Scharstein, Heiko Hirschmüller, York Kitajima, Greg Krathwohl, Nera Nešić, Xi Wang, and Porter Westling. High-resolution stereo datasets with subpixel-accurate ground truth. In *Pattern Recognition: 36th German Conference, GCPR 2014, Münster, Germany, September 2-5, 2014, Proceedings 36*, pages 31–42. Springer, 2014. 1, 2, 5, 7

[36] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, pages 746–760. Springer, 2012. 2, 5, 6

[37] Fabio Tosi, Yiyi Liao, Carolin Schmitt, and Andreas Geiger. Smd-nets: Stereo mixture density networks. In *CVPR*, pages 8942–8952, 2021. 1, 2, 3, 5, 6, 7, 8

[38] Lijun Wang, Jianming Zhang, Yifan Wang, Huchuan Lu, and Xiang Ruan. Cliffnet for monocular depth estimation with hierarchical embedding loss. In *ECCV*, pages 316–331. Springer, 2020. 2

[39] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE TIP*, 13(4):600–612, 2004. 5

[40] Ke Xian, Jianming Zhang, Oliver Wang, Long Mai, Zhe Lin, and Zhiguo Cao. Structure-guided ranking loss for single image depth prediction. In *CVPR*, pages 611–620, 2020. 2

[41] Guanglei Yang, Hao Tang, Mingli Ding, Nicu Sebe, and Elisa Ricci. Transformer-based attention networks for continuous pixel-wise prediction. In *ICCV*, pages 16269–16279, 2021. 2

[42] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *ICCV*, pages 3836–3847, 2023. 2, 6, 7

[43] Wenliang Zhao, Yongming Rao, Zuyan Liu, Benlin Liu, Jie Zhou, and Jiwen Lu. Unleashing text-to-image diffusion models for visual perception. *ICCV*, 2023. 2

[44] Zixiang Zhao, Jiangshe Zhang, Shuang Xu, Zudi Lin, and Hanspeter Pfister. Discrete cosine transform network for guided depth map super-resolution. In *CVPR*, pages 5697–5707, 2022. 2

[45] Zhiwei Zhong, Xianming Liu, Junjun Jiang, Debin Zhao, and Xiangyang Ji. Guided depth map super-resolution: A survey. *ACM Computing Surveys*, 2023. 2