
Roll2Depth: Zero-Shot Metric Depth Estimation Exploiting Camera’s Rolling Shutter Effect

Rui Xiao*

Shanghai University of Finance and Economics
MoE Key Laboratory of Interdisciplinary Research
of Computation and Economics
xiaorui@sufe.edu.cn

Jinming Xu*

Zhejiang University
jinming_xu@zju.edu.cn

Jinsong Han†

Zhejiang University
hanjinsong@zju.edu.cn

Abstract

Accurate monocular metric depth estimation, i.e., predicting absolute depth values, is critical for applications such as robotics and augmented reality. However, existing methods often suffer from poor transferability to in-the-wild scenarios due to the lack of knowledge of camera intrinsics. In this paper, we propose a novel framework, *Roll2Depth*, to recover metric scale in unseen environments with unknown camera parameters. Our key idea is to leverage scene-independent physical cues induced by the rolling shutter effect of standard CMOS sensors, together with an active light source, as metric priors. When a scene is illuminated by a flickering light, the rolling shutter produces distinctive stripe-like spatial patterns, revealing the scene’s shading with and without the light. This contrast encodes absolute depth cues that can be exploited for metric estimation. *Roll2Depth* isolates these patterns as metric priors to guide affine-invariant models on metric depth estimation in a zero-shot setting. Our evaluations on standard benchmark datasets, as well as a real-world dataset collected using commercial smartphones and our prototype system, demonstrate that *Roll2Depth*, as a low-cost active solution, outperforms state-of-the-art passive baselines in zero-shot settings. These results highlight its potential impact in applications such as indoor robotics and embodied perception, where affordable and effective active sensing would be valuable.

1 Introduction

Monocular metric depth estimation (MMDE) – predicting pixel-wise absolute depth values from a single RGB image – is a fundamental low-level vision task with wide-ranging applications in augmented reality [6], robotics [7], and autonomous driving [29, 50]. While recent architectural innovations achieve strong in-domain performance [11, 30, 33, 55], zero-shot MMDE in out-of-domain (i.e., in-the-wild) scenarios remains challenging [54, 56], as it depends not only on robust visual context perception, but also on accurate knowledge of camera intrinsics, such as focal length.

Given the challenges in zero-shot metric depth estimation, one line of research instead focuses on relative or affine-invariant depth estimation [19, 53], where depth is predicted up to an unknown global scale and shift. This paradigm benefits from large-scale pretraining on diverse datasets

*Equal contribution.

†Corresponding author

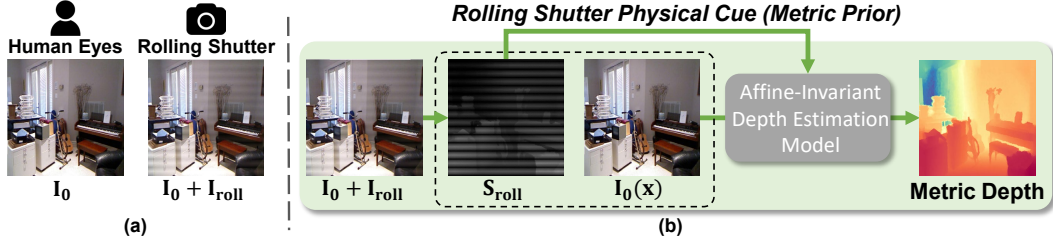


Figure 1: *Roll2Depth*’s high-level overview. (a) Rolling shutter cameras can capture light flicker – imperceptible to human eye – as stripe patterns in images. (b) *Roll2Depth* leverages the stripe-like shading, S_{roll} , which encodes absolute depth information, as scene-independent physical cues. After being isolated, S_{roll} is used as metric prior to an affine-invariant depth model, effectively guiding zero-shot monocular metric depth estimation across unseen environments and camera settings.

and demonstrates robust generalization. However, the lack of metric scale inevitably limits their applicability in metric-sensitive downstream applications. As a result, enabling zero-shot metric depth estimation remains a more desirable yet challenging objective. To this end, some recent methods incorporate camera metadata directly into the model to restore the metric scale [13, 35, 54]. These approaches have shown promising results in controlled settings where metadata is reliable and complete, while their performance can potentially degrade in more practical scenarios where metadata is noisy, incomplete, or entirely unavailable [56].

This leads us to the following research question: *Is there an alternative way to achieve zero-shot MMDE when camera metadata is noisy or unavailable?* To this end, we introduce a novel and orthogonal approach, *Roll2Depth*, which innovatively capitalizes on **scene-independent physical cues** – induced by the rolling shutter effect of common camera CMOS sensors, together with an active light source – as **metric priors**, eliminating the need for accurate camera metadata. A rolling shutter is a mechanism present in most commodity cameras, where the sensor captures the image row by row rather than all at once. When this row-wise exposure interacts with a flickering light source, it produces distinctive stripe-like spatial patterns with alternating bright and dark bands. These bands capture the scene’s shading both with and without the influence of the flickering light, as illustrated in Figure 1(a), revealing the relative contribution of the flickering light to the observed intensity. When normalized by the intensity ratio between the flickering source and ambient illumination, this relative shading is approximately associated with absolute depth via the inverse square law, thus providing a physically grounded metric prior. As shown in Figure 1(b), once isolated, this physical cue is used to complement the visual context and effectively guide *Roll2Depth*’s metric depth estimation process.

Grounded in this physical cue, we incorporate three key design innovations in *Roll2Depth*. First, the physical cue, i.e., the stripe pattern, is often subtle and entangled with complex scene content, and its isolation is an inherently ill-posed problem. To address this, we introduce a dedicated physical cue extraction network that learns to separate the stripe pattern from the entangled image. The second innovation lies in how this physical cue is incorporated. While the generation of stripe patterns can be described analytically, purely analytical approaches may fail to capture the variability and complexity of real-world scenes. Motivated by recent advances in accurate affine-invariant depth estimation [19, 53], we instead formulate the problem as predicting the global scale and shift needed to convert an affine-invariant depth map into metric depth. To estimate these parameters accurately, *Roll2Depth* employs a Vision Transformer (ViT)-based metric feature extractor together with a bin-based scalar regression module that provides stable and robust estimates. Third, to provide high-quality and diverse training data for *Roll2Depth*, we theoretically model the shading produced by the interaction between flickering illumination and the rolling shutter effect. This light modeling framework enables scalable data generation across diverse scenes and lighting conditions, and proves effective in practice, as demonstrated by *Roll2Depth*’s promising performance on real-world datasets.

By capitalizing on the rolling shutter effect, *Roll2Depth* generates these spatial patterns without the need for dedicated hardware, such as structured light projectors. It requires only a single flickering light source and is fully compatible with off-the-shelf camera hardware. We envision practical deployment of *Roll2Depth* in systems such as robotic, mobile, and embodied devices, where an active light source is either already present or easy to incorporate. For example, the flash on a smartphone or the front-facing illumination on an autonomous vehicle can be adapted to serve this purpose. This

approach is also highly cost-effective, making it well-suited for cost-sensitive platforms such as robot vacuums and consumer-grade drones. To summarize, our main contributions are:

- We are the first to introduce scene-independent physical cues induced by the rolling shutter as metric priors for zero-shot monocular depth estimation.
- We propose a complete framework to leverage the physical cues with: (i) a physical cue extraction network, (ii) a robust affine-to-metric transformation network, and (iii) scalable data synthesis with analytical light modeling.
- We build a hardware prototype to construct a real-world rolling shutter dataset for evaluation, demonstrating *Roll2Depth*'s effectiveness in real-world settings. We also release our code to facilitate future research.

2 Related Work

Depth Estimation with Zero-Shot Transferability. MMDE regresses dense depth from a single RGB image [9]. Although recent methods achieve solid in-domain performance [2, 34, 55], they often generalize poorly to unseen scenes due to dataset-, camera-, and geometry-specific priors. This has motivated zero-shot depth estimation, where models are evaluated on out-of-domain data without fine-tuning [4, 19, 35, 53, 56]. Existing approaches improve transferability either by predicting relative depth up to scale and shift [19, 21, 53], or by recovering metric scale through geometric pretraining, camera-aware fine-tuning, or test-time normalization using camera parameters [4, 13, 15, 32, 35, 54]. However, metric-depth methods remain sensitive to camera intrinsics and metadata quality, limiting their robustness in unconstrained settings such as in-the-wild videos and crowd-sourced images. In this paper, *Roll2Depth* proposes a new orthogonal path leveraging physical cues from the rolling shutter effect to enable transferable and physically grounded metric depth estimation.

Depth Estimation with Physical Cues. MMDE is inherently ill-posed, motivating prior work to incorporate physical cues such as focus/defocus and structured light. Focus-based methods estimate depth from lens-induced blur using focal stacks, sparse focal stacks, or single defocused images [10, 16, 18, 24–27, 39, 43, 48, 52]. However, they often require additional captures or all-in-focus references, which are difficult to obtain in dynamic scenes. Structured-light methods instead recover depth by projecting known patterns and analyzing their geometric distortions [17, 42, 51], but usually depend on dedicated projection hardware. Recent studies have explored incorporating physical or external sensing cues into affine-invariant depth models, such as Marigold [12, 44, 47]. *Roll2Depth* distinguishes itself by exploiting physical cues induced by the rolling shutter effect, enabling single-image metric depth estimation with standard CMOS sensors on commercial devices, without requiring focal stacks or specialized spatial projectors.

Rolling Shutter in Imaging and Sensing. Prior work has treated rolling-shutter artifacts as interference to be suppressed [37], or exploited them as an attack vector against vision systems [14]. Another line of computational imaging research leverages rolling-shutter exposure with synchronized illumination to enable epipolar and disparity gating for 3D reconstruction [28], ToF imaging [1], and selective imaging [45]. In contrast, *Roll2Depth* uses rolling shutter and flickering illumination as a single-shot temporal demultiplexer for depth cues. Note that *Roll2Depth* does not require synchronization between the camera and light source, thereby minimizing hardware overhead.

3 Depth Estimation with Rolling Shutter Prior

Roll2Depth is a depth estimation framework that leverages the rolling shutter-induced stripe pattern as a physically grounded metric prior. Our objective is to model the conditional distribution $D(\mathbf{d} \mid \mathbf{I})$, where $\mathbf{d} \in \mathbb{R}^{H \times W}$ is the target metric depth map and $\mathbf{I} \in \mathbb{R}^{H \times W \times 3}$ is the input image. The input image $\mathbf{I}$ contains embedded physical cues and is composed of two parts, i.e., $\mathbf{I} = \mathbf{I}_0 + \mathbf{I}_{\text{roll}}$, where $\mathbf{I}_0$ is the base RGB image captured under ambient illumination, and $\mathbf{I}_{\text{roll}}$ represents the stripe patterns induced by the interaction of a flickering light source with the rolling shutter readout. To further decompose the formation process, we express the image as:

$$\mathbf{I} = \mathbf{I}_0 + \mathbf{I}_{\text{roll}} = \mathbf{A}(\mathbf{S}_0 + \mathbf{S}_{\text{roll}}), \quad (1)$$

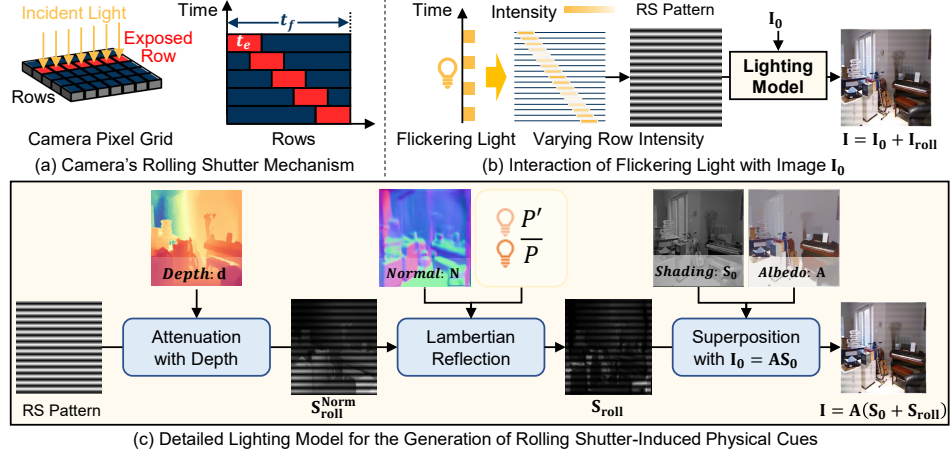


Figure 2: Generation of metric depth priors from the rolling shutter effect. (a) CMOS rolling shutter exposes the image row by row, acting as a sliding-window sampler. (b) The interaction between rolling shutter and high-frequency flickering illumination produces spatially periodic stripe-like patterns I_{roll} , which are superimposed onto the base image I_0 to form the observed image $I = I_0 + I_{\text{roll}}$. (c) Detailed lighting model: the RS pattern is depth-attenuated by d and undergoes a Lambertian reflection process, producing flicker-induced shading S_{roll} . This is modulated by scene albedo A and combined with ambient shading S_0 , resulting in the final image: $I = A(S_0 + S_{\text{roll}})$.

where A is the scene albedo, S_0 is the shading due to ambient light, and S_{roll} is the additional shading modulation caused by the flickering light. The component S_{roll} varies across rows due to the temporal offset of the rolling shutter, and serves as a depth-dependent cue that we aim to extract and exploit.

3.1 Metric Depth Prior Created by Rolling Shutter

Physical Cue Created by Rolling Shutter Distortion. Rolling shutter is a characteristic of CMOS sensors in commercial cameras. As shown in Figure 2(a), instead of exposing the entire frame simultaneously, rolling shutter sensors expose one row at a time, sequentially, to reduce caching overhead. The intensity of each row can be viewed as an integration or a moving average across the exposure duration t_e . Assuming sampling begins at time t , the intensity of row $A(t)$ is given by:

$$A(t) = \int_{-\infty}^{\infty} I_L(\tau) g_r(\tau - t) d\tau. \quad (2)$$

Here, $I_L(\tau)$ denotes the light intensity at time τ , and $g_r(\tau - t)$ is a rectangular gate function that equals 1 over $(t, t + t_e)$ and 0 otherwise. This sequential row-wise exposure mechanism interacts with flickering light to produce distinctive stripe-like patterns. Unlike uniform exposure, which results in a frame with uniform illumination, rolling shutter produces alternating bright and dark bands. These bands effectively encode the scene’s shading both with and without the influence of the flickering light within a single image. In contrast, uniform exposure requires multiple images captured under different lighting conditions to obtain equivalent information. As shown in Figure 2(b), the resulting rolling shutter (RS) pattern is then superimposed on the base image I_0 , yielding observed image $I = I_0 + I_{\text{roll}}$, where I_{roll} encodes the stripe-induced modulation tied to scene geometry and depth.

Metric Prior from the Physical Cue. As shown in Figure 2(c), under a flickering light source, the total shading can be decomposed into a static ambient component S_0 and a depth-sensitive flicker-induced component S_{roll} . The image formation model is:

$$I = A \cdot (S_0 + S_{\text{roll}}), \text{ with } S_{\text{roll}} = k \cdot \frac{\text{Stripe}}{d^2}, \quad (3)$$

where **Stripe** denotes the normalized rolling shutter (RS) pattern, d is the depth map, and k is an intensity scaling factor dependent on the power of the ambient light P and the flickering light P' . We define the normalized metric prior as $S_{\text{roll}}^{\text{norm}} = \frac{\text{Stripe}}{d^2}$, which encodes the inverse-square depth dependence and serves as the core physical prior in our method. This physically grounded

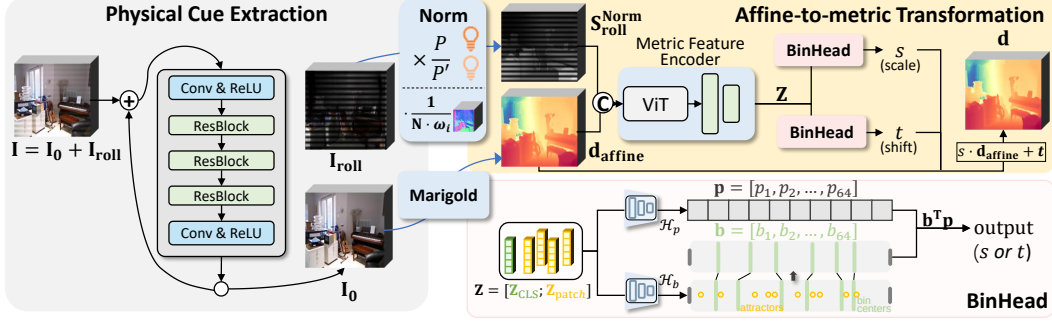


Figure 3: Figure depicts overview of *Roll2Depth*'s network architecture. *Roll2Depth* begins with a physical cue extraction module that decouples the input image into the rolling shutter-induced component $\mathbf{I}_{\text{roll}}$ and the clean image $\mathbf{I}_0$. The shading cue $\mathbf{S}_{\text{roll}}$ is then normalized using the relative light intensity $\frac{P}{P'}$ and corrected for surface orientation via $\frac{1}{\mathbf{N} \cdot \omega_i}$ to produce $\mathbf{S}_{\text{roll}}^{\text{norm}}$. This normalized physical cue, together with the affine-invariant depth $\mathbf{d}_{\text{affine}}$ from Marigold [19], is processed by a ViT-based metric feature encoder to obtain a compact feature $\mathbf{Z}$. Two bin-based scalar regression heads then predict the global scale and shift, which are applied to $\mathbf{d}_{\text{affine}}$ to obtain the metric depth $\mathbf{d}$.

formulation embeds an inverse-square depth dependence into the observed intensity. This signal is subsequently modulated by k and shaped by Lambertian surface reflectance, yielding the full shading component $\mathbf{S}_{\text{roll}}$. The complete image formation model incorporating reflectance can be expressed as:

$$\mathbf{I} = \mathbf{A} \cdot (\mathbf{S}_0 + \mathbf{S}_{\text{roll}}), \text{ with } \mathbf{S}_{\text{roll}} = k \cdot \frac{\text{Stripe}}{d^2} \cdot (\omega_i \cdot \mathbf{N}), \quad (4)$$

where ω_i is the incoming light directions, and $\mathbf{N}$ is the surface normal.

Takeaways. The rolling shutter-induced shading $\mathbf{S}_{\text{roll}}^{\text{norm}}$ encodes metric depth information, while also being modulated by surface normals and lighting, as shown in Equation 4. To serve as a reliable metric prior, it must be carefully extracted and effectively integrated. We describe how *Roll2Depth* is architected to achieve this in the following subsection.

3.2 Network Architecture Design

Our framework leverages the metric prior embedded in rolling-shutter cues to enable robust metric depth estimation. It consists of three main stages: (1) *physical cue extraction*, which extracts the physical cue $\mathbf{S}_{\text{roll}}^{\text{norm}}$ from the input image $\mathbf{I}$, also producing a clean image $\mathbf{I}_0$; (2) *affine-invariant depth estimation*, which obtains $\mathbf{d}_{\text{affine}}$ from $\mathbf{I}_0$ using a state-of-the-art pretrained depth estimator; and (3) *affine-to-metric transformation*, which predicts the global scale and shift that convert $\mathbf{d}_{\text{affine}}$ into the final metric depth $\mathbf{d}$. The third stage employs a ViT-based metric feature extractor followed by a specialized bin-based scalar regression module. We employ Marigold [19] as our affine-invariant depth estimator. The architecture overview is shown in Figure 3.

Physical Cue Extraction. To extract the physical cue $\mathbf{S}_{\text{roll}}^{\text{norm}}$ from the input image $\mathbf{I}$, we design a progressive physical cue extraction network f that predicts the clean image $\mathbf{I}_0$. The rolling shutter component is then recovered by subtraction, i.e., $\mathbf{I}_{\text{roll}} = \mathbf{I} - \mathbf{I}_0$. Extracting these subtle stripe patterns is challenging due to their low signal strength and strong dependency on scene illumination and geometry. To address this, we adopt a *progressive extraction* strategy by iteratively passing the input through an extraction block multiple times. Each extraction block contains two convolutional layers enclosing three intermediate residual blocks (ResBlocks). Each ResBlock has two 3×3 convolutional layers with 32 filters and ReLU activations, along with a skip connection that adds the block's input to its output. We formulate the training objective to minimize the dissimilarity between the predicted background $f(\mathbf{I})$ and the ground-truth ambient image $\mathbf{I}_0$, considering both spatial and frequency domain discrepancies. Motivated by the **row-wise periodicity** of rolling-shutter-induced patterns, we introduce a frequency-domain loss to complement the spatial reconstruction loss (see Appendix F for an ablation study). The loss function $\mathcal{L}_{\text{ext}}$ is defined as:

$$\mathcal{L}_{\text{ext}} = \|\mathbf{f}(\mathbf{I}) - \mathbf{I}_0\|_2 + \|F(\mathbf{f}(\mathbf{I})) - F(\mathbf{I}_0)\|_2, \quad (5)$$

where $\|\cdot\|_2$ denotes the Euclidean norm and F represents the 1D discrete Fourier transform along the vertical direction. After obtaining $\mathbf{I}_{\text{roll}}$, we estimate the albedo $\mathbf{A}$ and surface normal $\mathbf{N}$ using existing methods [22, 54], and normalize it with power factor k , which can be obtained through calibrated camera measurements, e.g., under fixed or normalized exposure, or measured directly using photodiodes in practical deployments. We then divide out these components to compute the normalized metric prior $\mathbf{S}_{\text{roll}}^{\text{norm}}$.

Affine-to-metric Transformation. Given the normalized physical cue $\mathbf{S}_{\text{roll}}^{\text{norm}}$ and the clean image $\mathbf{I}_0$, *Roll2Depth* first estimates a high-quality affine-invariant depth map $\mathbf{d}_{\text{affine}}$ from $\mathbf{I}_0$ using a state-of-the-art estimator [19]. The objective of *Roll2Depth* is then to model the conditional depth distribution $D(\mathbf{d} \mid \mathbf{d}_{\text{affine}}, \mathbf{S}_{\text{roll}}^{\text{norm}})$, which captures how the metric prior guides the recovery of true scale. Specifically, *Roll2Depth* learns to anchor $\mathbf{d}_{\text{affine}}$ in the metric space by predicting the global scale s and shift t that transform the affine-invariant depth into metric depth, expressed as $\mathbf{d} = s \cdot \mathbf{d}_{\text{affine}} + t$, thereby producing metrically accurate depth maps guided by the extracted physical prior.

Metric Feature Encoder. To achieve this, we concatenate $\mathbf{S}_{\text{roll}}^{\text{norm}}$ and $\mathbf{d}_{\text{affine}}$ and feed them into a metric feature encoder that outputs $\mathbf{Z} = [\mathbf{Z}_{\text{global}}; \mathbf{Z}_{\text{patch}}] \in \mathbb{R}^{577 \times 128}$, where $\mathbf{Z}_{\text{global}}$ denotes a global token summarizing the entire image and $\mathbf{Z}_{\text{patch}}$ represents 576 patch tokens over a 24×24 spatial grid. This encoding is produced by an encoder composed of a pretrained ViT [8] and a trainable two-layer multilayer perceptron (MLP). The ViT extracts 1024-dimensional features for both global and patch tokens, which are subsequently reduced to 128 dimensions by the MLP to form the compact feature representation $\mathbf{Z}$. The feature $\mathbf{Z}$ is then fed into two BinHeads for scale and shift prediction.

Bin-based Scalar Regression. Feature $\mathbf{Z}$ is then processed by two identical BinHeads, which predict the global scale s and the shift t , respectively. Our design takes inspiration from LocalBins [3], which predicts pixel-wise bin centers with their corresponding probabilities for stable depth estimates. While we do not perform pixel-wise prediction, we adopt this bin-based formulation for scalar regression due to its stability and robustness. Each BinHead first predicts 64 bin centers $\mathbf{b} = [b_1, \dots, b_{64}]$ through a subnetwork $\mathcal{H}_b$, which first estimates 64 coarse bin centers from the global token using a two-layer MLP, and then refines them with the patch tokens via an attractor module inspired by [4]. For each patch token, an attractor MLP outputs an adjustment point $a_k (k = 1, \dots, 576)$. These attractors pull each bin center toward feature-informed positions according to

$$b'_i = b_i + \Delta b_i, \quad \Delta b_i = \sum_{k=1}^{576} (a_k - b_i) \cdot e^{-\alpha |a_k - b_i|^\gamma}, \quad (6)$$

where α and γ are hyperparameters controlling attraction strength and decay.

To obtain the final metric depth prediction, the refined bin centers are linearly combined, weighted by their predicted probability scores. Building on the order-aware probability formulation in [4], $\mathcal{H}_p$ processes the token sequence $\mathbf{Z}$ through a learned projection and a LogBinomial module, yielding the bin probability vector $\mathbf{p} = [p_1, \dots, p_{64}]$. The final scalar is computed as the expectation $\mathbf{b}^\top \mathbf{p}$, yielding either the global scale s or the shift t . These two scalars define the affine transformation $\mathbf{d} = s \cdot \mathbf{d}_{\text{affine}} + t$, thereby converting the affine-invariant depth into the final metric depth.

Training-Time Affine Variation. During training, instead of obtaining $\mathbf{d}_{\text{affine}}$ from Marigold, we generate it directly from the ground-truth metric depth $\mathbf{d}$ for efficiency. We first apply a percentile-based affine normalization. For each depth map, we compute its 2nd and 98th percentiles, $\mathbf{d}_2$ and $\mathbf{d}_{98}$, and acquire $\mathbf{d}_{\text{affine}} = \frac{\mathbf{d} - \mathbf{d}_2}{\mathbf{d}_{98} - \mathbf{d}_2}$. To simulate the variability of affine-invariant estimators, we then map $\mathbf{d}_{\text{affine}}$ to a random interval of the form $[\text{rand}(-n, -m), \text{rand}(m, n)]$, where $0 < m \leq n$. (The standard $[-1, 1]$ normalization corresponds to the special case $m = n = 1$.) We adopt $m = 0.8$ and $n = 1.2$ in our implementation. This flexible range improves robustness to the scale and shift fluctuations exhibited by affine-invariant depth estimators during inference.

4 Experiments

4.1 Experimental setup

Benchmark Datasets. We train *Roll2Depth* on **NYU Depth v2** [40], which provides densely annotated indoor RGB-D images, and evaluate zero-shot generalization on four unseen benchmarks: **SUN RGB-D** [41], **iBims** [20], **DIODE indoor** [46], and **HyperSim** [38], which vary in scene layout, lighting conditions, and camera characteristics.



Figure 4: Figure depicts the collection setup for depth dataset with real rolling shutter effects. **Left:** monocular RGB images are captured using a smartphone, and ground-truth depth is provided by an Intel RealSense D455 stereo camera. We calibrate the relative pose between the two cameras and remap the depth map to the smartphone’s viewpoint. **Right:** due to the smartphone’s limited flashlight control, we design a custom active lighting system using a 5W LED driven by an Arduino Uno. The LED is positioned near the phone’s flash to simulate a flash-aligned flickering light source for generating rolling shutter stripe patterns.

Self-Collected Depth Dataset with Rolling Shutter Effects. To evaluate *Roll2Depth* under real-world rolling shutter conditions, we collect a real-world dataset containing 647 paired samples. Each sample includes a depth map, an RGB image with rolling shutter cues, and the corresponding clean RGB image. Figure 4 shows our data collection setup. We capture monocular images using a commercial smartphone with the *mcpro24fps* app, setting the exposure time to 1/2000 seconds with auto ISO. Images are captured at 1920×1080 and resized to 640×480 to match *Roll2Depth*’s input resolution. For flickering illumination, we are limited by the smartphone’s inability to programmatically control its flashlight. To address this, we built a custom active lighting system using a low-power 5W LED controlled by an Arduino Uno board. The Arduino generates square-wave signals via pulse-width modulation (PWM) at a 50% duty cycle, with user-defined flicker frequencies of 1 KHz. The PWM output drives a MOSFET switch to produce stable high-frequency flickering illumination. The LED is positioned near the phone’s flashlight to simulate a flash-like light origin aligned with the camera axis. **Ground-truth** depth is obtained using an Intel RealSense D455 stereo camera [36]. We calibrate the relative pose between the smartphone camera and the RealSense RGB camera, and warp the RealSense depth maps into the smartphone view. The stereo camera also provides clean monocular images for evaluating baseline methods.

Evaluation Metrics. We evaluate in metric depth space $\mathbf{d}$ using three standard metrics following prior works [2, 31, 55]: the absolute relative error (AbsRel), defined as $\frac{1}{M} \sum_{i=1}^M |\mathbf{d}_i - \hat{\mathbf{d}}_i| / \mathbf{d}_i$; the root mean squared error (RMSE), computed as $\left(\frac{1}{M} \sum_{i=1}^M (\mathbf{d}_i - \hat{\mathbf{d}}_i)^2 \right)^{0.5}$; and the threshold accuracy δ_1 , given by $\frac{1}{M} \sum_{i=1}^M \left[\max \left(\frac{\mathbf{d}_i}{\hat{\mathbf{d}}_i}, \frac{\hat{\mathbf{d}}_i}{\mathbf{d}_i} \right) < 1.25 \right]$. Here, $\hat{\mathbf{d}}$ is the predicted depth map, $\mathbf{d}$ is the ground-truth depth map, and M is the number of valid pixels.

Implementation Details. Our framework is implemented in PyTorch. We train the model using the AdamW optimizer [23] with $\beta_1 = 0.9$, $\beta_2 = 0.999$, a weight decay of 0.1, and a learning rate of 1×10^{-5} . The metric feature encoder is initialized from the pretrained DINOv2-L ViT model [8] and jointly fine-tuned with the rest of the network. We use a batch size of 2 and an input resolution of 640×480 unless otherwise noted. Training proceeds for 26 epochs on a single NVIDIA RTX 3090 GPU, with each epoch taking approximately 2 hours.

4.2 Results on Standard Benchmark with Synthetic Rolling Shutter Cue

We first evaluate the zero-shot MMDE on SUN RGB-D [41], DIODE Indoor [46], HyperSim [38], and iBims-1 [20]. State-of-the-art many-shot and zero-shot passive baselines are evaluated using clean images without RS cues. For *Roll2Depth*, we synthetically inject rolling shutter patterns to simulate the RS-based metric prior³. Quantitative results are shown in Table 1. *Roll2Depth* achieves consistently strong performance, ranking first on all four benchmarks. Therefore, ***Roll2Depth*, as a low-cost active system, outperforms state-of-the-art passive baselines.** Compared with them, *Roll2Depth* shows more consistent generalization across diverse benchmarks. This suggests that

³We further validate this evaluation protocol in **Appendix C** by analyzing the noise and uncertainty in real-world captures.

Table 1: Zero-shot monocular depth estimation results on four benchmark datasets. **Bold** indicates the best performance, underline the second best, and *italic* the third best. We re-evaluate the results using a consistent pipeline to ensure fair and comprehensive metric comparison. "N/A" indicates that the method is trained on that dataset and therefore not applicable for evaluation on it.

Method	SUN RGB-D			DIODE Indoor			HyperSim			iBims-1		
	$\delta_1 \uparrow$	AbsRel $\downarrow$	RMSE $\downarrow$	$\delta_1 \uparrow$	AbsRel $\downarrow$	RMSE $\downarrow$	$\delta_1 \uparrow$	AbsRel $\downarrow$	RMSE $\downarrow$	$\delta_1 \uparrow$	AbsRel $\downarrow$	RMSE $\downarrow$
AdaBins [2]	0.771	0.159	0.476	0.174	0.443	1.963	0.221	0.483	6.546	0.555	0.212	0.901
LocalBins [3]	0.777	<i>0.156</i>	0.470	0.229	0.412	1.853	0.234	0.468	6.362	0.558	0.211	0.880
NeWCRFs [55]	0.798	<u>0.151</u>	<i>0.424</i>	0.187	0.404	1.867	0.255	0.442	6.017	0.548	0.206	0.861
ZoeDepth [4]	0.864	0.119	0.346	0.376	0.327	1.588	0.292	0.410	5.771	0.658	0.169	0.711
ZeroDepth [13]	0.199	0.624	0.889	0.432	0.300	1.392	0.350	0.417	5.882	0.746	0.164	0.634
Metric3D [54]	0.703	0.367	0.813	0.508	0.384	1.439	0.371	0.389	4.775	0.530	0.279	1.274
UniDepth-C [35]	0.400	0.436	0.583	0.628	0.238	0.968	<u>0.513</u>	<u>0.313</u>	3.789	0.811	<i>0.148</i>	<i>0.536</i>
UniDepth-V [35]	0.269	0.493	0.668	0.798	0.181	<i>0.760</i>	<i>0.501</i>	<i>0.332</i>	<i>3.871</i>	0.234	0.357	1.063
MoGe-2 [49]	0.722	0.379	0.462	0.557	<i>0.200</i>	0.933	N/A	N/A	N/A	<i>0.826</i>	<u>0.137</u>	<u>0.510</u>
DepthPro [5]	0.468	0.474	0.631	0.689	0.230	<i>0.738</i>	N/A	N/A	N/A	<u>0.837</u>	0.162	<u>0.510</u>
Roll2Depth (Ours)	0.867	0.171	0.317	0.848	0.147	0.593	0.572	0.264	<i>4.540</i>	0.921	0.094	0.406

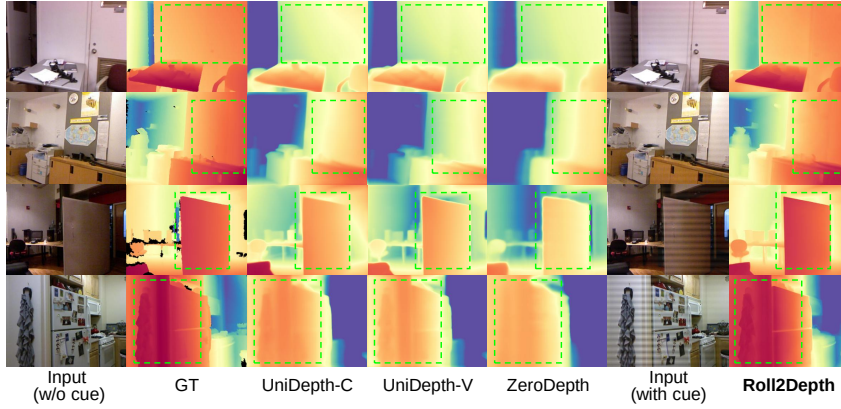


Figure 5: Zero-shot qualitative MMDE results. Each pair of consecutive rows corresponds to one test sample. Green dashed boxes highlight regions where baseline methods exhibit *global metric shifts*, while *Roll2Depth* effectively corrects them by leveraging physically grounded cues.

incorporating explicit physical cues provides valuable metric information beyond purely learned scale priors or camera metadata, which may be noisy or unavailable in real-world settings. This observation is further supported by the qualitative comparisons in Figure 5, where *Roll2Depth* better preserves the correct metric scale. By guiding an affine-invariant estimator with metric priors from physical cues, *Roll2Depth* achieves robust generalization across diverse scenes and camera conditions.

4.3 Results on Self-Collected Dataset with Real Rolling Shutter Cue

Table 2: Zero-shot monocular depth estimation results on our collected datasets with real rolling shutter cues, as well as on four benchmark datasets. **Bold** indicates the best performance, underline the second best, and *italic* the third best. Baseline methods were evaluated using ground truth clean images without stripe patterns.

Method	Scene 1					Scene 2				
	$\delta_1 \uparrow$	$\delta_2 \uparrow$	$\delta_3 \uparrow$	RMSE $\downarrow$	AbsRel $\downarrow$	$\delta_1 \uparrow$	$\delta_2 \uparrow$	$\delta_3 \uparrow$	RMSE $\downarrow$	AbsRel $\downarrow$
ZoeDepth [4]	<u>0.627</u>	<u>0.723</u>	<u>0.776</u>	<u>1.472</u>	2.827	0.341	0.730	0.955	1.076	0.473
ZeroDepth [13]	0.259	0.651	0.742	1.655	3.364	0.569	0.776	0.923	0.977	0.455
Metric3D [54]	0.546	0.643	0.703	<i>1.508</i>	<u>2.462</u>	0.092	0.288	0.621	2.218	0.620
UniDepth-C [35]	0.160	0.661	0.736	1.798	3.478	<u>0.618</u>	<u>0.857</u>	0.955	<u>0.881</u>	<i>0.428</i>
UniDepth-V [35]	0.424	0.692	0.754	1.625	3.176	<i>0.587</i>	0.808	0.955	0.868	0.433
MoGe-2 [49]	<i>0.573</i>	<i>0.717</i>	<i>0.772</i>	1.608	2.866	0.485	<i>0.841</i>	0.960	0.909	<u>0.411</u>
DepthPro [5]	0.073	0.503	0.702	2.351	3.977	0.374	0.767	0.936	1.420	0.728
Roll2Depth (Ours)	0.629	0.732	0.795	1.183	2.460	0.769	0.924	<u>0.957</u>	0.885	0.407

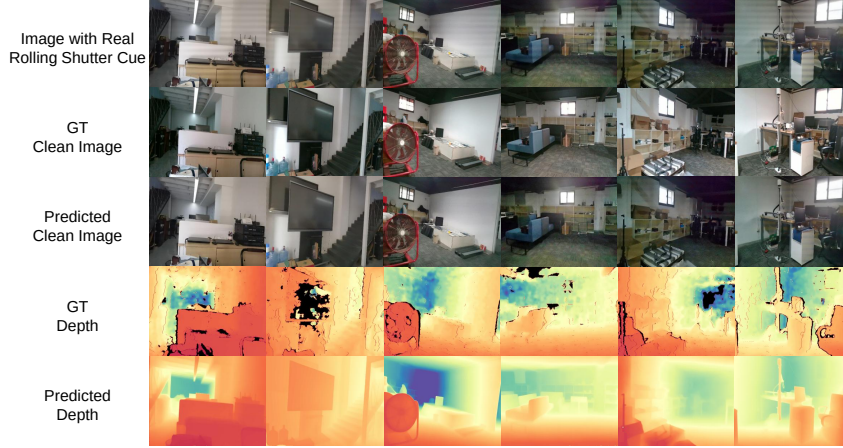


Figure 6: Zero-shot qualitative MMDE results on our dataset with real rolling shutter cues. Line 3 shows the clean image restored by *Roll2Depth* after extracting the physical cue. This clean image, together with the extracted cue, is then used for metric depth estimation, shown in Line 5.

To evaluate *Roll2Depth*'s performance under real rolling shutter conditions, we construct a real-world dataset with RS cues, as detailed in §4.1. Baselines are evaluated on ground-truth clean images without stripe patterns, while *Roll2Depth* operates on images containing real RS-induced physical cues. We report standard metrics (δ_1 , AbsRel, and RMSE) as well as extended ones (δ_2 and δ_3). Quantitative results are shown in Table 2, with qualitative results shown in Figure 6. *Roll2Depth* achieves the best or competitive performance across most metrics in both scenes, demonstrating its practical effectiveness under real imaging conditions. These results confirm that the RS-induced cues can be reliably extracted from real camera captures and transformed into effective metric priors. This further shows that ***Roll2Depth* is not only theoretically grounded but also deployable in real-world environments.** We position *Roll2Depth* as a **compelling alternative to passive state-of-the-art solutions** in scenarios where low-cost active illumination is available, such as indoor robotic platforms. By introducing a lightweight active sensing cue, *Roll2Depth* achieves improved metric depth estimation with minimal hardware overhead. Please refer to **Appendix D** for a detailed discussion of *Roll2Depth*'s positioning.

4.4 Ablation Study

Table 3 evaluates the contribution of each component introduced in §3. Removing physical cue extraction, i.e., directly using $\mathbf{I}$ in place of $\mathbf{S}_{\text{roll}}^{\text{norm}}$, significantly degrades performance across all datasets. Similar drops are observed when disabling bin-based regression module or training-time affine variation. Detailed ablations on the contribution of the rolling-shutter physical cue and each module are present in **Appendix A and B**, demonstrating the effectiveness of *Roll2Depth*'s design.

Table 3: Ablation study of *Roll2Depth*'s modules. Underline values indicate the full *Roll2Depth* design achieves the best performance.

Method	SUN RGB-D		DIODE		HyperSim		iBims-1	
	$\delta_1 \uparrow$	RMSE $\downarrow$	$\delta_1 \uparrow$	RMSE $\downarrow$	$\delta_1 \uparrow$	RMSE $\downarrow$	$\delta_1 \uparrow$	RMSE $\downarrow$
Full (all modules)	<u>0.867</u>	<u>0.317</u>	<u>0.848</u>	0.593	0.572	4.540	<u>0.921</u>	<u>0.406</u>
– physical cue extraction	0.794	0.395	0.694	0.892	0.462	5.100	0.836	0.577
– bin-based regression	0.841	0.322	0.716	0.939	0.450	5.331	0.838	0.557
– affine variation	0.832	0.340	0.844	0.586	0.588	4.451	0.918	0.411

5 Discussion and Conclusion

Limitation and Deployment Considerations. In this paper, we primarily focus on validating *Roll2Depth* in indoor scenarios. While outdoor deployment, such as on vehicles, is not explored, it

remains a feasible direction, particularly given the availability of high-powered front-facing lights. However, future work must investigate the effective sensing range and model the more directional nature of such lighting to ensure sufficient illumination and reliable stripe pattern formation under varying outdoor conditions. Looking forward, we envision integrating *Roll2Depth* into indoor commercial devices such as service robots, robotic vacuum cleaners, and mixed reality headsets to enhance their 3D perception with a low-cost, robust solution. In the longer term, deploying *Roll2Depth* in outdoor systems could be explored by modulating vehicular light sources and studying their interaction with real-world driving environments.

Extension of Uncertainty-Aware Affine-to-Metric Transformation. The reliability of extracted physical cues may degrade under challenging conditions, such as non-Lambertian reflectance (e.g., specular surfaces) or weak flickering signals due to attenuation and strong ambient illumination (see efficacy curves in Appendix E). A promising direction is to incorporate an uncertainty head into the physical cue extractor to predict per-pixel confidence, which could adaptively reweight ViT tokens or attention to suppress unreliable regions during global scale and shift estimation. This mechanism could also mitigate dependency on external pipelines by reducing the impact of inaccurate intermediate outputs. We leave this extension for future work.

Broader impacts. This work points to a broader direction in leveraging low-cost hardware to complement passive MMDE approaches. When capture-time control is available, **this hardware-enhanced paradigm could be highly valuable**, as evidenced by LiDAR-, radar-, and structured-light-based solutions used in robotics, AR/VR applications, autonomous driving, and 3D reconstruction. Compared to them, *Roll2Depth* offers a *low-cost alternative* that operates with commercial cameras.

Conclusion. We present *Roll2Depth*, a novel framework for zero-shot metric monocular depth estimation that leverages the rolling shutter-induced stripe pattern as a physically grounded metric prior. *Roll2Depth* effectively learns to disentangle these metric priors and thus transform affine-invariant depth into accurate metric depth, without requiring fine-tuning or camera calibration at test time. Our extensive evaluation, including real-world experiments, demonstrates that *Roll2Depth*, as a low-cost active system, outperforms state-of-the-art passive baselines in zero-shot settings.

Acknowledgments and Disclosure of Funding

We sincerely thank our anonymous reviewers for their valuable feedback. This paper is supported by the National Key R&D Program of China (2023YFA1009500), the National Natural Science Foundation of China (NSFC) under grants No.62372400 and 62602402, and the Fundamental Research Funds for the Central Universities (2025110528-0).

References

- [1] Supreeth Achar, Joseph R. Bartels, William L. Whittaker, Kiriakos N. Kutulakos, and Srinivasa G. Narasimhan. Epipolar time-of-flight imaging. *ACM Trans. Graph.*, 36(4):37:1–37:8, 2017.
- [2] Shariq Farooq Bhat, Ibraheem Alhashim, and Peter Wonka. Adabins: Depth estimation using adaptive bins. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 4009–4018. IEEE, 2021.
- [3] Shariq Farooq Bhat, Ibraheem Alhashim, and Peter Wonka. Localbins: Improving depth estimation by learning local distributions. In *European Conference Computer Vision, ECCV*, volume 13661, pages 480–496. Springer, 2022.
- [4] Shariq Farooq Bhat, Reiner Birkel, Diana Wofk, Peter Wonka, and Matthias Müller. Zoedepth: Zero-shot transfer by combining relative and metric depth. *CoRR*, abs/2302.12288, 2023.
- [5] Aleksei Bochkovskii, Amaël Delaunoy, Hugo Germain, Marcel Santos, Yichao Zhou, Stephan R Richter, and Vladlen Koltun. Depth pro: Sharp monocular metric depth in less than a second. *arXiv preprint arXiv:2410.02073*, 2024.
- [6] Kangle Deng, Andrew Liu, Jun-Yan Zhu, and Deva Ramanan. Depth-supervised nerf: Fewer views and faster training for free. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 12872–12881. IEEE, 2022.

- [7] Xingshuai Dong, Matthew A. Garratt, Sreenatha G. Anavatti, and Hussein A. Abbass. Towards real-time monocular depth estimation for robotics: A survey. *IEEE Trans. Intell. Transp. Syst.*, 23(10):16940–16961, 2022.
- [8] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations, ICLR*. OpenReview.net, 2021.
- [9] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In *Annual Conference on Neural Information Processing Systems, NIPS*, pages 2366–2374, 2014.
- [10] Paolo Favaro, Andrea Mennucci, and Stefano Soatto. Observing shape from defocused images. *Int. J. Comput. Vis.*, 52(1):25–43, 2003.
- [11] Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, and Dacheng Tao. Deep ordinal regression network for monocular depth estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 2002–2011. IEEE, 2018.
- [12] Jakub Gregorek and Lazaros Nalpantidis. Steeredmarigold: Steering diffusion towards depth completion of largely incomplete depth maps. In *IEEE International Conference on Robotics and Automation, ICRA*, pages 13304–13311. IEEE, 2025.
- [13] Vitor Guizilini, Igor Vasiljevic, Dian Chen, Rares Ambrus, and Adrien Gaidon. Towards zero-shot scale-aware monocular depth estimation. In *IEEE/CVF International Conference on Computer Vision, ICCV*, pages 9199–9209. IEEE, 2023.
- [14] Dongfang Guo and Rui Tan. Rolling in the deep: Exploiting rolling shutter effect against stereo depth estimation in drones. In Ardalán Amiri Sani, Chenren Xu, Przemysław Pawełczak, and Swarun Kumar, editors, *Proceedings of the 23rd Annual International Conference on Mobile Systems*, pages 705–710. ACM, 2025.
- [15] Yuliang Guo, Sparsh Garg, S. Mahdi H. Miangoleh, Xinyu Huang, and Liu Ren. Depth any camera: Zero-shot metric depth estimation from any camera. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 26996–27006. Computer Vision Foundation / IEEE, 2025.
- [16] Shir Gur and Lior Wolf. Single image depth estimation trained via depth from defocus cues. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 7683–7692. IEEE/CVF, 2019.
- [17] Xuanlun Huang, Chenyang Wu, Xiaolan Xu, Baishun Wang, Sui Zhang, Chihchiang Shen, Chiennan Yu, Jiaying Wang, Nan Chi, Shaohua Yu, et al. Polarization structured light 3d depth image sensor for scenes with reflective surfaces. *Nature Communications*, 14(1):6855, 2023.
- [18] Hayato Ikoma, Cindy M. Nguyen, Christopher A. Metzler, Yifan Peng, and Gordon Wetzstein. Depth from defocus with learned optics for imaging and occlusion-aware depth estimation. In *IEEE International Conference on Computational Photography, ICCP*, pages 1–12. IEEE, 2021.
- [19] Bingxin Ke, Anton Obukhov, Shengyu Huang, Nando Metzger, Rodrigo Caye Daudt, and Konrad Schindler. Repurposing diffusion-based image generators for monocular depth estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 9492–9502. IEEE, 2024.
- [20] Tobias Koch, Lukas Liebel, Friedrich Fraundorfer, and Marco Körner. Evaluation of cnn-based single-image depth estimation methods. In *European Conference Computer Vision (ECCV) Workshops*, volume 11131, pages 331–348. Springer, 2018.
- [21] Hsin-Ying Lee, Hung-Yu Tseng, Hsin-Ying Lee, and Ming-Hsuan Yang. Exploiting diffusion prior for generalizable dense prediction. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 7861–7871. IEEE, 2024.
- [22] Zhengqin Li, Mohammad Shafiei, Ravi Ramamoorthi, Kalyan Sunkavalli, and Manmohan Chandraker. Inverse rendering for complex indoor scenes: Shape, spatially-varying lighting and SVBRDF from a single image. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 2472–2481. IEEE, 2020.

- [23] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR*, 2019.
- [24] Yawen Lu, Garrett Milliron, John Slagter, and Guoyu Lu. Self-supervised single-image depth estimation from focus and defocus clues. *IEEE Robotics Autom. Lett.*, 6(4):6281–6288, 2021.
- [25] Junjie Luo, John Mamish, Alan Fu, Thomas Concannon, Josiah D. Hester, Emma Alexander, and Qi Guo. Focal split: Untethered snapshot depth from differential defocus. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 26965–26974. Computer Vision Foundation / IEEE, 2025.
- [26] Maxim Maximov, Kevin Galim, and Laura Leal-Taixé. Focus on defocus: Bridging the synthetic to real domain gap for depth estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 1068–1077. IEEE, 2020.
- [27] Michael Möller, Martin Benning, Carola Schönlieb, and Daniel Cremers. Variational depth from focus reconstruction. *IEEE Trans. Image Process.*, 24(12):5369–5378, 2015.
- [28] Matthew O’Toole, Supreeth Achar, Srinivasa G. Narasimhan, and Kiriakos N. Kutulakos. Homogeneous codes for energy-efficient illumination and imaging. *ACM Trans. Graph.*, 34(4):35:1–35:13, 2015.
- [29] Dennis Park, Rares Ambrus, Vitor Guizilini, Jie Li, and Adrien Gaidon. Is pseudo-lidar needed for monocular 3d object detection? In *2021 IEEE/CVF International Conference on Computer Vision, ICCV*, pages 3122–3132. IEEE, 2021.
- [30] Vaishakh Patil, Christos Sakaridis, Alexander Liniger, and Luc Van Gool. P3depth: Monocular depth estimation with a piecewise planarity prior. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 1600–1611. IEEE, 2022.
- [31] Suraj Patni, Aradhye Agarwal, and Chetan Arora. Ecodepth: Effective conditioning of diffusion models for monocular depth estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 28285–28295, 2024.
- [32] Luigi Piccinelli, Christos Sakaridis, Mattia Segù, Yung-Hsu Yang, Siyuan Li, Wim Abbeloos, and Luc Van Gool. Unik3d: Universal camera monocular 3d estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 1028–1039. Computer Vision Foundation / IEEE, 2025.
- [33] Luigi Piccinelli, Christos Sakaridis, and Fisher Yu. idisc: Internal discretization for monocular depth estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 21477–21487. IEEE, 2023.
- [34] Luigi Piccinelli, Christos Sakaridis, and Fisher Yu. idisc: Internal discretization for monocular depth estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 21477–21487. IEEE, 2023.
- [35] Luigi Piccinelli, Yung-Hsu Yang, Christos Sakaridis, Mattia Segù, Siyuan Li, Luc Van Gool, and Fisher Yu. Unidepth: Universal monocular metric depth estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 10106–10116. IEEE, 2024.
- [36] RealSense. Intel realsense depth camera d455. <https://www.intelrealsense.com/depth-camera-d455/>, 2024.
- [37] Erik Ringaby and Per-Erik Forssén. Scan rectification for structured light range sensors with rolling shutters. In *IEEE International Conference on Computer Vision, ICCV*, pages 1575–1582. IEEE Computer Society, 2011.
- [38] Mike Roberts, Jason Ramapuram, Anurag Ranjan, Atulit Kumar, Miguel Ángel Bautista, Nathan Paczan, Russ Webb, and Joshua M. Susskind. Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding. In *IEEE/CVF International Conference on Computer Vision, ICCV*, pages 10892–10902. IEEE, 2021.
- [39] Haozhe Si, Bin Zhao, Dong Wang, Yunpeng Gao, Mulin Chen, Zhigang Wang, and Xuelong Li. Fully self-supervised depth estimation from defocus clue. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 9140–9149. IEEE, 2023.
- [40] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from RGBD images. In *European Conference on Computer Vision, ECCV*, volume 7576, pages 746–760. Springer, 2012.

- [41] Shuran Song, Samuel P. Lichtenberg, and Jianxiong Xiao. SUN RGB-D: A RGB-D scene understanding benchmark suite. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 567–576. IEEE, 2015.
- [42] Varun Sundar, Sizhuo Ma, Aswin C. Sankaranarayanan, and Mohit Gupta. Single-photon structured light. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 17844–17854. IEEE, 2022.
- [43] Supasorn Suwajanakorn, Carlos Hernández, and Steven M. Seitz. Depth from focus with your mobile phone. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 3497–3506. IEEE, 2015.
- [44] Chinmay Talegaonkar, Nikhil Gandudi Suresh, Zachary Novack, Yash Belhe, Priyanka Nagasamudra, and Nicholas Antipa. Repurposing marigold for zero-shot metric depth estimation via defocus blur cues. In *Annual Conference on Neural Information Processing Systems, NeurIPS*, 2025.
- [45] Tomoki Ueda, Hiroyuki Kubo, Suren Jayasuriya, Takuya Funatomi, and Yasuhiro Mukaigawa. Slope disparity gating using a synchronized projector-camera system. In *IEEE International Conference on Computational Photography, ICCP*, pages 1–9. IEEE, 2019.
- [46] Igor Vasiljevic, Nicholas I. Kolkin, Shanyi Zhang, Ruotian Luo, Haochen Wang, Falcon Z. Dai, Andrea F. Daniele, Mohammadreza Mostajabi, Steven Basart, Matthew R. Walter, and Gregory Shakhnarovich. DIODE: A dense indoor and outdoor depth dataset. *CoRR*, abs/1908.00463, 2019.
- [47] Massimiliano Viola, Kevin Qu, Nando Metzger, Bingxin Ke, Alexander Becker, Konrad Schindler, and Anton Obukhov. Marigold-dc: Zero-shot monocular depth completion with guided diffusion. In *IEEE/CVF International Conference on Computer Vision, ICCV*, pages 5359–5370. IEEE, 2025.
- [48] Ning-Hsu (Albert) Wang, Ren Wang, Yu-Lun Liu, Yu-Hao Huang, Yu-Lin Chang, Chia-Ping Chen, and Kevin Jou. Bridging unsupervised and supervised depth from focus via all-in-focus supervision. In *International Conference on Computer Vision, ICCV*, pages 12601–12611. IEEE, 2021.
- [49] Ruicheng Wang, Sicheng Xu, Yue Dong, Yu Deng, Jianfeng Xiang, Zelong Lv, Guangzhong Sun, Xin Tong, and Jiaolong Yang. Moge-2: Accurate monocular geometry with metric scale and sharp details. *arXiv preprint arXiv:2507.02546*, 2025.
- [50] Yan Wang, Wei-Lun Chao, Divyansh Garg, Bharath Hariharan, Mark E. Campbell, and Kilian Q. Weinberger. Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 8445–8453. IEEE, 2019.
- [51] Xianmin Xu, Yuxin Lin, Haoyang Zhou, Chong Zeng, Yaxin Yu, Kun Zhou, and Hongzhi Wu. A unified spatial-angular structured light for single-view acquisition of shape and reflectance. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 206–215, 2023.
- [52] Fengting Yang, Xiaolei Huang, and Zihan Zhou. Deep depth from focus with differential focus volume. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 12632–12641. IEEE/CVF, 2022.
- [53] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 10371–10381. IEEE, 2024.
- [54] Wei Yin, Chi Zhang, Hao Chen, Zhipeng Cai, Gang Yu, Kaixuan Wang, Xiaozhi Chen, and Chunhua Shen. Metric3d: Towards zero-shot metric 3d prediction from A single image. In *IEEE/CVF International Conference on Computer Vision, ICCV*, pages 9009–9019, 2023.
- [55] Weihao Yuan, Xiaodong Gu, Zuozhuo Dai, Siyu Zhu, and Ping Tan. Neural window fully-connected crfs for monocular depth estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 3906–3915. IEEE, 2022.
- [56] Yizhou Zhao, Hengwei Bian, Kaihua Chen, Pengliang Ji, Liao Qu, Shao-yu Lin, Weichen Yu, Haoran Li, Hao Chen, Jun Shen, Bhiksha Raj, and Min Xu. Metric from human: Zero-shot monocular metric depth estimation via test-time adaptation. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2024.

A Evidence of Rolling Shutter Effectiveness via Ablation

To further ablate the contribution of the rolling-shutter physical cue, we keep the full architecture but replace the input with the clean image $\mathbf{I}_0$, removing the rolling-shutter cue. The model is then retrained under this setting. The results in Table 4 demonstrate that the **rolling-shutter cue is indispensable for improved metric performance**. The architecture alone is insufficient to achieve the performance gains, as removing this cue leads to a clear drop.

Table 4: Ablation study on *Roll2Depth*’s rolling shutter cue.

Method	SUN RGB-D		DIODE		HyperSim		iBims-1	
	AbsRel↓	RMSE↓	AbsRel↓	RMSE↓	AbsRel↓	RMSE↓	AbsRel↓	RMSE↓
w/o cue	1.604	2.302	0.712	1.843	0.520	5.435	0.624	1.455
with cue	<u>0.171</u>	<u>0.317</u>	<u>0.147</u>	<u>0.593</u>	<u>0.264</u>	<u>4.540</u>	<u>0.094</u>	<u>0.406</u>

B Ablation Study for Each Module

We provide a detailed ablation study for each module of *Roll2Depth*.

Bin-based Scalar Regression. Table 5 shows detailed ablation results. Compared with two baseline designs – (i) global regression, which directly regresses the global scale and shift from ViT features using MLPs, and (ii) patch-weighted averaging, which predicts per-patch scale and shift followed by confidence-weighted fusion, our bin-based formulation achieves consistently higher accuracy across all benchmarks. On average, it improves δ_1 by 0.11 and reduces RMSE by 0.364.

Backbone of Metric Feature Extractor. Table 5 compares different backbone choices for the metric feature extractor. Among the three variants, ViT-L achieves the best overall performance, while BEiT and Swin Transformer remain competitive. We select ViT as our default encoder because it offers strong representational capacity, and its global attention mechanism aligns well with the requirements of our bin-based regression module.

Training-Time Affine Variation. Table 6 evaluates the effect of training-time affine variation by sweeping the parameters m and n . We also compare against the standard setting, which excludes this module entirely. The results show that sampling the affine range within $m = 0.8$ and $n = 1.2$ yields the best performance, and we therefore adopt these values in all experiments.

Table 5: Ablation on *Roll2Depth*’s affine-to-metric transformation, comparing scalar regression strategy and backbone variants.

Method	SUN RGB-D		DIODE		HyperSim		iBims-1	
	$\delta_1 \uparrow$	RMSE↓	$\delta_1 \uparrow$	RMSE↓	$\delta_1 \uparrow$	RMSE↓	$\delta_1 \uparrow$	RMSE↓
<i>Scalar Regression Strategy</i>								
global regression	0.841	0.322	0.716	0.939	0.450	5.331	0.838	0.557
patch-weighted regression	0.771	0.378	0.680	1.003	0.422	5.474	0.803	0.621
bin-based regression	<u>0.867</u>	<u>0.317</u>	<u>0.848</u>	<u>0.593</u>	<u>0.572</u>	<u>4.540</u>	<u>0.921</u>	<u>0.406</u>
<i>Backbone of Metric Feature Encoder</i>								
BEiT-Large	0.860	0.305	0.847	0.588	0.596	4.449	0.919	0.399
Swin-Transformer	0.843	0.321	0.802	0.662	0.527	4.652	0.880	0.449
ViT-L (DINOv2)	<u>0.867</u>	<u>0.317</u>	<u>0.848</u>	<u>0.593</u>	<u>0.572</u>	<u>4.540</u>	<u>0.921</u>	<u>0.406</u>

Table 6: Ablation on Training-Time Affine Variation. Results under different random scaling ranges applied to training inputs.

Method	SUN RGB-D		DIODE		HyperSim		iBims-1	
	$\delta_1 \uparrow$	RMSE↓	$\delta_1 \uparrow$	RMSE↓	$\delta_1 \uparrow$	RMSE↓	$\delta_1 \uparrow$	RMSE↓
w/o variation	0.832	0.340	0.844	0.586	0.588	4.45	0.918	0.411
m=0.9 n=1.1	0.820	0.350	0.822	0.625	0.565	4.696	0.897	0.446
m=0.7 n=1.3	0.851	0.324	0.815	0.651	0.582	4.467	0.901	0.433
m=0.8 n=1.2	<u>0.867</u>	<u>0.317</u>	<u>0.848</u>	<u>0.593</u>	<u>0.572</u>	<u>4.540</u>	<u>0.921</u>	<u>0.406</u>

C Robustness to Synthesis-Model Mismatch under Real-World Reflectance Deviations

In the standard benchmark evaluation (§4.2), rolling shutter cues are synthetically generated because existing depth benchmarks do not provide images captured under flickering illumination. Since *Roll2Depth* is trained with synthetic RS cues based on a Lambertian image formation model, a natural concern is whether the observed gains rely on an exact match between the training and testing synthesis pipelines. To examine this, we conduct a model-mismatched evaluation by introducing **untrained non-Lambertian reflectance components at test time**. Specifically, we extend the Lambertian shading model to a Blinn–Phong form, adding specular residuals that are absent during training:

$$\mathbf{S}'_{\text{roll}} = \mathbf{S}_{\text{roll}} + k_s \mathbf{S}_{\text{roll}}^{\text{NL}} = k \cdot \frac{\text{Stripe}}{d^2} (\omega_i \cdot \mathbf{N} + k_s (\mathbf{h} \cdot \mathbf{N})^n).$$

Here, $\mathbf{h}$ denotes the half-vector, $\mathbf{n}$ is a shininess map controlling the specular lobe width, and k_s controls the relative specular strength. Following the Blinn–Phong model, k_s is typically small (0.05–0.2), while larger values ($k_s \gtrsim 0.3$) correspond to highly specular materials like glasses.

This setting deliberately breaks the exact correspondence between the training-time Lambertian synthesis and the test-time rolling shutter cue generation. It therefore evaluates whether *Roll2Depth* can still exploit the underlying depth-dependent RS structure when the observed cue is perturbed by realistic reflectance deviations. As shown in Table 7, *Roll2Depth* remains robust across different levels of non-Lambertian perturbation, despite being trained only with Lambertian RS cues. This suggests that *Roll2Depth* does not merely overfit to a matched synthetic pipeline, but can still extract useful metric priors under reflectance mismatch. This evaluation shows that while synthetic RS cues cannot capture all real-world imaging complexities, the conclusion drawn in §4.2 remains meaningful for validating *Roll2Depth*. Together with the real-capture evaluation in §4.3, these results support the effectiveness of *Roll2Depth* in practical RS settings.

Table 7: Robustness against synthesis-model mismatch under real-world reflectance deviations.

Method	SUN RGB-D		DIODE		HyperSim		iBims-1	
	$\delta_1 \uparrow$	RMSE $\downarrow$	$\delta_1 \uparrow$	RMSE $\downarrow$	$\delta_1 \uparrow$	RMSE $\downarrow$	$\delta_1 \uparrow$	RMSE $\downarrow$
w/o Noise	0.867	0.317	0.848	0.593	0.572	4.540	0.921	0.406
$k_s = 0.1$	0.866	0.316	0.848	0.603	0.561	4.573	0.917	0.414
$k_s = 0.3$	0.860	0.321	0.831	0.636	0.535	4.643	0.910	0.443
$k_s = 0.5$	0.848	0.334	0.818	0.675	0.502	4.713	0.891	0.481

D Position of *Roll2Depth*: An Active, Effective, and Low-Cost System

Roll2Depth is a hardware-assisted MMDE system that leverages a simple active light source to provide additional physical cues. Therefore, its comparison with purely passive zero-shot MMDE methods should be interpreted carefully: while *Roll2Depth* achieves better performance than existing passive baselines, we do not position it as a direct state-of-the-art replacement for passive methods, since it operates under a different sensing setting. At the same time, *Roll2Depth* is also fundamentally different from traditional LiDAR- or radar-based depth sensing systems, which rely on dedicated and more sophisticated hardware. Instead, *Roll2Depth* occupies a practical middle ground: it introduces minimal hardware assistance but effectively improves metric depth estimation, while remaining lightweight and compatible with cost-sensitive platforms such as smartphones, robot vacuums, and consumer-grade drones. To modulate an existing light source, *Roll2Depth* only requires a commodity MOSFET-based switching circuit, which costs only a few dollars. We hope this design point encourages further exploration of lightweight hardware-assisted depth estimation systems that bridge passive vision and dedicated active sensing.

E Efficacy Curves under Distance and Ambient Light Variations

We evaluate the robustness under varying distances and ambient illumination conditions. Table 8 reports the performance of *Roll2Depth* across different sensing distances. The results show that *Roll2Depth* maintains relatively stable performance from 1 m to 10 m, with AbsRel ranging from

Table 8: Performance under varying sensing distances.

Distance (m)	0-1	1-2	2-3	3-4	4-5	5-6	6-7	7-8	8-9	9-10
AbsRel ↓	1.364	0.170	0.221	0.174	0.181	0.187	0.190	0.241	0.199	0.181

Table 9: Performance under varying active-to-ambient SNR. Lower SNR indicates stronger ambient illumination relative to the active signal.

SNR (dB)	-21	-20	-19	-18	-17	-16	-15	-14	-13	-12	-11	-10	-9	-8
AbsRel ↓	0.336	0.389	0.365	0.345	0.331	0.409	0.507	0.470	0.444	0.515	0.363	0.349	0.439	0.448

0.170 to 0.241. Performance degrades more noticeably at distances below 1 m, where depth estimation becomes particularly sensitive to prediction errors. We further investigate the impact of ambient illumination by varying the active-to-ambient signal-to-noise ratio (SNR), which characterizes the relative strength of the active illumination signal against ambient light. A lower SNR indicates stronger ambient illumination relative to the active signal. As shown in Table 9, *Roll2Depth* achieves AbsRel values ranging from 0.331 to 0.515 across SNR levels from -21 to -8 dB, without exhibiting a sharp performance degradation threshold. For reference, an SNR of approximately -20 dB corresponds to a 10 W LED under 2,000 lux ambient illumination in our experimental setup. The robustness to variations in sensing distance and ambient illumination can be attributed to the progressive physical cue extraction network, which progressively isolates subtle rolling-shutter stripe patterns even when the active illumination signal is attenuated.

Table 10: Ablation study on the frequency loss.

Setting	Scene 1		Scene 2	
	AbsRel ↓	RMSE ↓	AbsRel ↓	RMSE ↓
w/o frequency loss	4.489	2.177	0.625	1.090
w/ frequency loss	2.460	1.183	0.407	0.885

F Ablation Study on the Frequency Loss

We conduct an ablation study to evaluate the contribution of the frequency loss term in Eq. (5). As shown in Table 10, removing the frequency loss consistently degrades depth estimation accuracy across both scenes. The frequency loss is motivated by the row-wise periodic structure of rolling-shutter stripe patterns along the vertical image axis. Such periodic structures can be effectively characterized using a 1D Fourier transform along the vertical direction. By penalizing discrepancies in the frequency domain, the frequency loss complements the spatial reconstruction loss and helps suppress residual or distorted stripe components. This improves the fidelity of the extracted physical cues, thereby contributing to more accurate metric depth estimation.

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: Answer: [Yes].

Justification: We discuss the main contribution and scope in abstract and Section 1.

Guidelines:

- The answer [N/A] 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 [N/A] 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: Answer: [Yes]

Justification: We discussion the limitation in Section 5.

Guidelines:

- The answer [N/A] 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: [N/A]

Justification: There are no theoretical results in our paper.

Guidelines:

- The answer [N/A] 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 provide detailed experimental settings and system implementation for our main experiment in Section 4.1.

Guidelines:

- The answer [N/A] 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?

Answer: [Yes]

Justification: Our code and dataset are available in the supplementary material.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://neurips.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://neurips.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, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: We provide detailed experimental settings and system implementation for our main experiment in Section 4.1.

Guidelines:

- The answer [N/A] 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: Error bars were not included; however, we report results using widely accepted and representative metrics for depth estimation, consistent with standard practice in the field.

Guidelines:

- The answer [N/A] 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.)
- 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 provide information on the computer resources in Section 4.1.

Guidelines:

- The answer [N/A] 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: The paper conforms to the NeurIPS Code of Ethics.

Guidelines:

- The answer [N/A] 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 discuss broader impacts in Section 5.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] 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.

- 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., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: This paper poses no such risks.

Guidelines:

- The answer [N/A] 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: We cite all used public datasets and pre-trained models in Section 4.1.

Guidelines:

- The answer [N/A] 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.

- 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: [N/A]

Justification: We do not introduce new assets in the paper.

Guidelines:

- The answer [N/A] 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: [N/A]

Justification: This paper does not involve crowdsourcing or research with human subjects.

Guidelines:

- The answer [N/A] 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: [N/A]

Justification: This paper does not involve research with human subjects.

Guidelines:

- The answer [N/A] 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.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.