CVPR
#20845

M2Depth: Unifying Monocular Depth Foundation Priors with Multi-View Stereo

Anonymous CVPR submission

Paper ID 20845

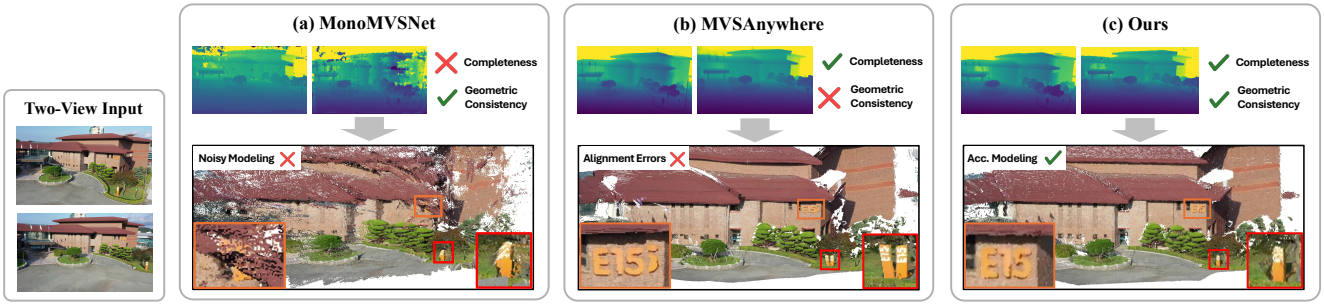


Figure 1. 3D reconstruction comparison of DFM-based MVS methods in a sparse-view setting. (a) The MVS-centric method (MonoMVSNet [21]) fails to produce a clean or generalizable reconstruction. (b) The DFM-centric method (MVSAnywhere [18]) generates a clean depth map but lacks geometric consistency, suffering from scale ambiguity and alignment errors. (c) In contrast, our method produces both a clean depth map and an accurate, geometrically aligned 3D structure.

Abstract

Deep learning-based Multi-View Stereo (MVS) has advanced significantly but often generalizes poorly to unseen scenes, particularly in occluded areas or regions with limited view overlap. To mitigate this, recent approaches integrate Depth Foundation Models (DFMs) into MVS pipelines to provide monocular depth priors. However, existing methods typically rely on a static, one-way fusion scheme, which fails to fully exploit the complementary strengths of both modalities. We propose a novel framework that overcomes this limitation by tightly coupling a DFM with a cascade MVS pipeline through a bidirectional mutual refinement strategy. Our method leverages MVS depth to resolve the scale ambiguity in monocular predictions, while the monocular depth, in turn, enhances the structural completeness and fine-grained detail of the MVS estimate. Furthermore, we introduce a prior-guided cost volume refinement mechanism that effectively integrates multi-view and monocular information via attention-based fusion and discretized depth bins, thereby promoting local geometric consistency. Extensive experiments demonstrate that our approach outperforms state-of-the-art MVS methods on benchmark datasets, producing highly complete and generalizable depth maps with sharp object boundaries, even in sparse-view cases.

1. Introduction

Multi-View Stereo (MVS) aims to reconstruct detailed 3D geometries by estimating dense pixel correspondences across images captured from multiple viewpoints. Recent deep learning-based approaches [17, 19, 42] generate robust features and utilize end-to-end optimization frameworks, significantly outperforming traditional MVS methods [4, 13]. Despite these advances, MVS approaches often struggle to generalize well to unseen scene structures, particularly under complex geometries, viewpoint variations, and challenging lighting conditions. Additionally, accuracy significantly deteriorates in regions with occlusions or limited view overlap, where depth estimation becomes inherently challenging due to ambiguous scene geometry.

Concurrently, Vision Foundation Models (VFMs) [2, 30, 31] have shown impressive generalization capabilities across diverse visual tasks. Among these, Depth Foundation Models (DFMs), such as DepthAnythingV2 [41], trained on extensive image-depth datasets, yield robust monocular depth predictions even in previously unseen domains, generating sharp and clear depth maps from single images. However, their monocular nature leads to scale ambiguity and a lack of explicit multi-view consistency, thereby limiting their direct application to accurate 3D reconstruction.

To overcome these limitations, two primary approaches

CVPR #20845

have recently been proposed to integrate DFMs [41] with MVS. The first approach (e.g., MonoMVSNet [21]) is an MVS-centric method that incorporates a monocular prior into an MVS pipeline. This prior provides structural guidance in depth boundary regions, which improves depth estimation accuracy. The second approach (e.g., MVSAnywhere [18]) employs a DFM-centric pipeline that leverages multi-view information as a prior. It combines multi-view cost volumes with a pre-trained DFM encoder to generate more complete and scale-aware depth maps.

Despite these performance gains, both lines of work share a fundamental limitation: they treat the complementary modality as a static, one-way prior, fusing it without a feedback mechanism. This prevents each stream from dynamically compensating for the other's inherent limitations. Consequently, MVS-centric pipelines still struggle to produce clean, generalizable depth in textureless or occluded regions, while DFM-centric pipelines remain prone to scale ambiguity, yielding misaligned geometry. Moreover, this unidirectional fusion causes errors from an inaccurate prior to propagate without correction through subsequent stages.

In this paper, we propose a novel framework that tightly couples a DFM with a cascade MVS pipeline [7], moving beyond the conventional, static one-way fusion paradigm [18, 21]. Our approach introduces a co-refinement strategy, ensuring that both monocular and MVS depth representations are progressively improved throughout the cascade. This method operates iteratively and bidirectionally, where the monocular and MVS depths mutually reference and complement one another. Specifically, the MVS depth is leveraged to improve the local scale consistency of the monocular depth, while the monocular depth provides a structural prior to enhance the details and completeness of the MVS depth.

Additionally, we propose a cost volume refinement mechanism that effectively fuses multi-view and monocular information. Our method constructs a monocular cost volume and integrates it with the multi-view cost volume via an attention-based mechanism. We also discretize the monocular depth into depth bins that act as structural priors, promoting local consistency within the fused volume. Consequently, our approach generates highly complete depth maps, particularly in challenging regions with occlusions or limited view overlap. As illustrated in Fig. 1, unlike existing methods [18, 21], our method produces both clean depth maps and an accurate, geometrically aligned 3D structure, demonstrating its robust integration of DFM and MVS strengths.

Our main contributions can be summarized as follows:

- We introduce a novel MVS framework that integrates monocular priors from DFMs, enabling clean and complete depth maps in challenging cases while ensuring accurate 3D reconstruction.
- We propose a bidirectional mutual refinement mechanism that leverages MVS depth to align monocular scale and, in turn, uses monocular depth to enhance MVS structure.
- We propose a prior depth-guided cost volume refinement strategy, fusing monocular and multi-view volumes using bin masks to improve spatial consistency.
- Our method achieves high accuracy and strong generalization performance across various benchmarks [1, 23, 34], surpassing existing MVS approaches in both depth estimation and 3D reconstruction. Source code for our method is publicly available.[1]

2. Related Work

2.1. Multi-View Stereo

Recent deep learning-based MVS approaches [19, 42] have significantly improved reconstruction accuracy. MVSNet [42], for example, introduced an end-to-end pipeline consisting of feature extraction, cost volume construction, and cost volume regularization. Despite achieving high accuracy, MVSNet uses computationally expensive 3D convolutions, leading to substantial GPU memory requirements and limited scalability to high-resolution inputs.

Subsequent works proposed cascade cost-volume frameworks [10, 17, 39] to alleviate this issue. These methods progressively refine depth hypotheses across multiple stages, reducing computational complexity by narrowing the search space at each stage. Further enhancements have introduced local and global attention mechanisms [12, 24, 28] to refine feature representations effectively. Geometry-aware methods, such as GeoMVSNet [46] and GoMVS [37], explicitly incorporate structural information and surface normals to improve precision and consistency in depth estimation. Transformer-based approaches, such as MVSFormer [6], have emerged to improve feature extraction capabilities. An enhanced version, MVSFormer++ [7], introduces side view attention, refined attention scaling, and advanced positional encoding to reinforce inter-view fusion and consistency.

2.2. Depth Foundation Model

Vision Foundation Models (VFMs) [41] are models trained on extensive datasets, providing strong generalization to various visual tasks with little to no fine-tuning. In monocular depth estimation, Depth Foundation Models (DFMs) have recently gained attention. Dense Prediction Transformer [32] leverages transformer architectures to enhance dense prediction tasks with improved structural and contextual understanding. DepthAnything [40, 41] further exploits web-scale image-depth datasets to enable robust, zero-shot depth estimation across diverse scenes. However, these monocular models inherently suffer from scale ambiguity, which hinders their direct application for 3D modeling.

Recent research [3, 20, 36, 47] has incorporated DFMs into stereo depth estimation frameworks, exploiting their

[1] https://github.com/released-after-acceptance

CVPR
#20845

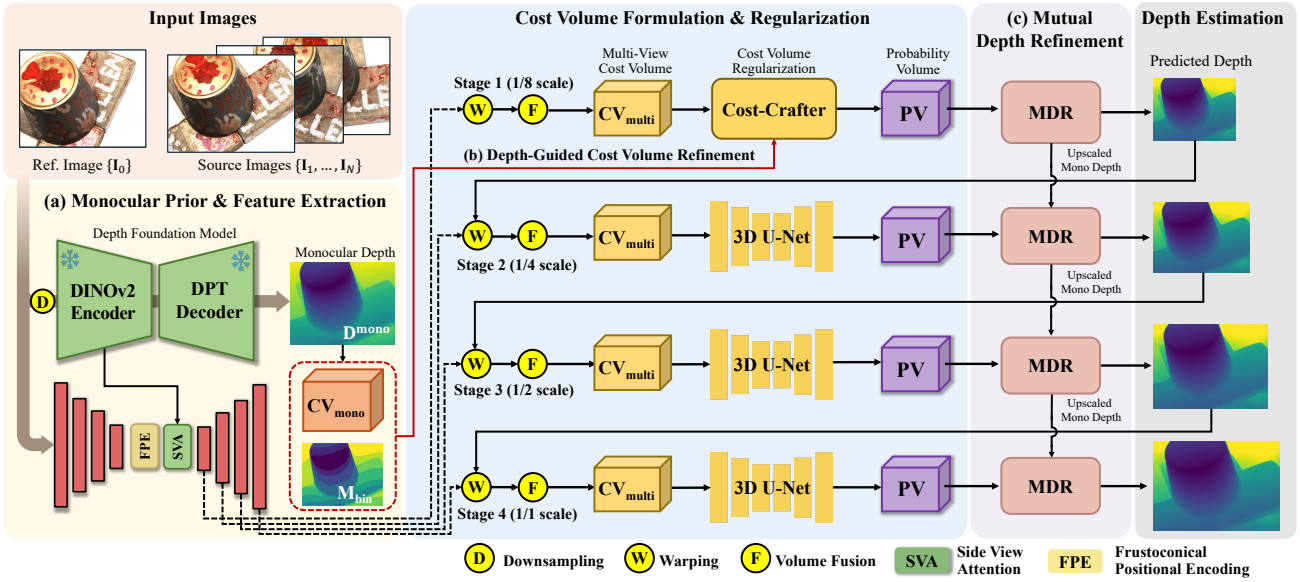


Figure 2. Our system follows a cascaded framework. Similar to the baseline [7], we construct and regularize a multi-view cost volume $\mathbf{CV}_{\text{multi}}$ at each stage, which is then converted into a probability volume **PV**. The MVS depth map $\mathbf{D}^{\text{mvs}}$ is then estimated from **PV** via depth expectation. Unlike the baseline, our method (a) extracts a monocular depth map $\mathbf{D}^{\text{mono}}$ using the DFM [41], (b) refines the cost volume in the first stage using $\mathbf{D}^{\text{mono}}$, and (c) performs iterative mutual refinement between the MVS depth and the monocular depth at all stages.

strong generalization and semantic priors. MonSter [9] proposes a framework to effectively fuse monocular depth priors into the stereo matching pipeline. FoundationStereo [36] introduces a zero-shot stereo approach directly integrating monocular depth priors without requiring fine-tuning.

While many studies [3, 20, 36, 47] focus on stereo vision, fewer efforts [18, 21] have integrated DFMs within MVS. MonoMVSNet [21] is an MVS-centric approach injecting a monocular prior into its cost volume to guide and enhance stereo matching. MVSAnywhere [18] adopts a DFM-centric pipeline, using multi-view cost volumes as conditioning for a large monocular model to generate scale-aware predictions. However, MonoMVSNet relies on a static prior, making it susceptible to error propagation when the initial depth is inaccurate. Conversely, MVSAnywhere's loose coupling with multi-view geometry can lead to misaligned reconstructions.

In contrast, our method uses a bidirectional co-refinement strategy where MVS and monocular estimates mutually correct each other iteratively. This tight coupling resolves DFM scale ambiguity while enhancing MVS completeness, avoiding error propagation and geometric misalignment.

3. Method

Our objective is to estimate a high-quality depth map $\mathbf{D}_0$ for a reference image $\mathbf{I}_0 \in \mathbb{R}^{3\times H\times W}$, given N source images $\{\mathbf{I}_i\}_{i=1}^{N}$ and their corresponding camera parameters. As illustrated in Fig. 2, our framework integrates a Depth Foundation Model (DFM), specifically DepthAnythingV2 [40], into a cascade MVS pipeline [17].

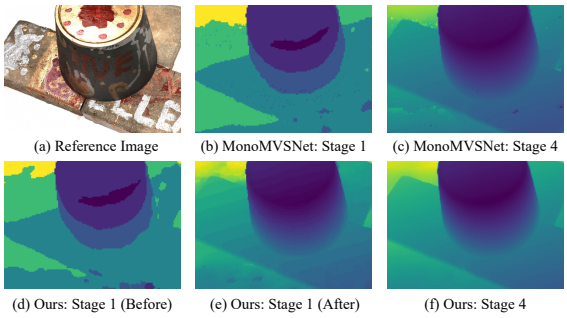
The core MVS pipeline employs a four-stage coarse-to-fine approach, progressively refining depth estimates from $1/8$ to full resolution. To achieve this, it extracts multi-scale visual features $\{\mathbf{F}^l_{\text{FPN}}\}_{l=1}^{4} \in \mathbb{R}^{C_l\times H_l\times W_l}$ using a feature pyramid network (FPN) [25]. At each stage l, stereo matching is performed using the corresponding features $\mathbf{F}^l_{\text{FPN}}$.

Concurrently, the DFM estimates a monocular depth prior $\mathbf{D}^{\text{mono}} \in \mathbb{R}^{H_1\times W_1}$ for the reference image $\mathbf{I}_0$, where $H_1 = \frac{H}{8}$ and $W_1 = \frac{W}{8}$. This predicted depth map $\mathbf{D}^{\text{mono}}$ provides structural priors and fine-grained shape cues. Therefore, it is used to guide the depth refinement process at each cascade stage, enhancing the completeness of the final estimation.

To effectively leverage this prior depth information, we introduce two key contributions:

1. **Prior-Guided Cost Volume Refinement** (Sect. 3.1): Applied at the initial cascade stage, this mechanism enhances the multi-view cost volume. It integrates a monocular cost volume (derived from $\mathbf{D}^{\text{mono}}$) and structural bin masks to establish a robust geometric foundation for the subsequent stages.
2. **Mutual Depth Refinement** (Sect. 3.2): This module operates across all stages, performing mutually complementary updates. It simultaneously refines the MVS depth $\mathbf{D}^{\text{mvs}}$ to enhance fine-grained structural details, while also using the multi-view geometric cues to achieve pixel-accurate scale alignment for the monocular depth $\mathbf{D}^{\text{mono}}$.

By integrating monocular priors, which provide global structure, with MVS features that offer geometric precision, our method generates highly accurate and detailed depth



(a) Reference Image (b) MonoMVSNet: Stage 1 (c) MonoMVSNet: Stage 4
(d) Ours: Stage 1 (Before) (e) Ours: Stage 1 (After) (f) Ours: Stage 4

Figure 3. Depth map results at Stage 1 and 4. (a) Reference image. MonoMVSNet [21] may produce a noisy Stage 1 depth map (b), and these artifacts propagate to its final Stage 4 result (c). While our initial Stage 1 depth (d) is similarly noisy, our mutual refinement module (e) effectively cleans the depth map, enhancing object boundaries and surface consistency. This provides a robust foundation, leading to a much cleaner final depth map (f).

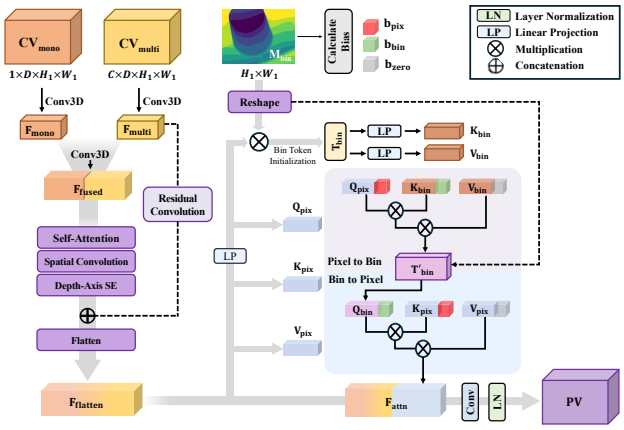


Figure 4. Architecture of the Cost-Crafter. This module enhances the multi-view cost volume by incorporating a monocular cost volume and structural bin masks, which serve as spatial priors.

maps. Unlike prior approaches [18, 21], we jointly upsample and refine the MVS and monocular depths at each stage. As shown in Fig. 3, this iterative co-refinement process progressively enhances the scale consistency of the monocular prior, ensuring robust performance even when the initial priors from the DFM are significantly mis-scaled or noisy.

3.1. Prior Depth-Guided Cost Volume Refinement

We propose a cost volume refinement mechanism that leverages structural depth priors to fuse multi-view and monocular cost volumes. This mechanism is applied exclusively at the initial cascade stage, where computational efficiency and global geometric reasoning are most critical for establishing a robust foundation for subsequent refinement.

First, we construct the multi-view cost volume $\mathbf{CV}_{\text{multi}} \in \mathbb{R}^{C\times D\times H_1\times W_1}$ by performing stereo matching across multi-view features, where C and D are the channel and depth dimensions. Concurrently, a monocular cost volume $\mathbf{CV}_{\text{mono}} \in \mathbb{R}^{1\times D\times H_1\times W_1}$ is created using soft one-hot encoding from the prior depth map. We also generate a depth bin mask $M_{\text{bin}} \in \{1,\dots,M\}^{H_1\times W_1}$ by discretizing the continuous depth prior into M uniformly spaced bins, assigning each pixel an index. These three components, $\mathbf{CV}_{\text{multi}}$, $\mathbf{CV}_{\text{mono}}$, and $\mathbf{M}_{\text{bin}}$, serve as inputs to our Cost-Crafter module.

Cost-Crafter module. Fig. 4 shows the architecture of the Cost-Crafter module. Both cost volumes $\mathbf{CV}_{\text{multi}}$ and $\mathbf{CV}_{\text{mono}}$ are projected via 3D convolutions, yielding intermediate features $\mathbf{F}_{\text{multi}}$ and $\mathbf{F}_{\text{mono}}$. These two features are concatenated along the channel dimension and then fused by another 3D convolution, producing the fused volume $\mathbf{F}_{\text{fused}} \in \mathbb{R}^{C\times D\times H_1\times W_1}$.

To enrich this fused volume with global context, we apply a self-attention mechanism to $\mathbf{F}_{\text{fused}}$. The refined 3D volume is flattened across the depth and spatial dimensions to produce pixel-level features $\mathbf{F}_{\text{flatten}} \in \mathbb{R}^{S\times C}$, where $S = D \times H_1 \times W_1$.

To provide a compact global depth representation that complements pixel-level features, we first derive representative bin features from the prior-guided depth bins. This step enables the network to capture depth-aware global structure that cannot be obtained from pixel-level features alone. To compute representative bin features, the bin mask $\mathbf{M}_{\text{bin}}$ is first converted into a one-hot representation $\mathbf{M}_{\text{bin}}^{\text{one-hot}} \in \{0,1\}^{S\times M}$, which is then reshaped and expanded along the depth dimension for subsequent pixel-to-bin attention. A matrix multiplication between the transpose of $\mathbf{M}_{\text{bin}}^{\text{one-hot}}$ and $\mathbf{F}_{\text{flatten}}$ is performed, followed by averaging over the pixels within each bin, yielding aggregated bin tokens $\mathbf{T}_{\text{bin}} \in \mathbb{R}^{M\times C}$, which serve as representative bin-level features that encode pixels belonging to the same depth bin.

The module employs a bidirectional cross-attention mechanism between the pixel-level features $\mathbf{F}_{\text{flatten}}$ and the aggregated bin tokens $\mathbf{T}_{\text{bin}}$. This interaction utilizes a shared cross-attention module, $\text{Attn}(\mathbf{Q},\mathbf{K},\mathbf{V})$, which incorporates scalar positional bias terms ($\mathbf{b}_{\text{pix}}$, $\mathbf{b}_{\text{bin}}$) into the standard cross-attention score.

In the first pass **(pixel-to-bin)**, the pixel features query the bin tokens. This allows each pixel to gather context from the global depth structure:

$$\mathbf{F}_{\text{pix}\to\text{bin}} = \text{Attn}\big(\text{LP}_Q(\mathbf{F}_{\text{flatten}}),\ \text{LP}_K(\mathbf{T}_{\text{bin}}),\ \text{LP}_V(\mathbf{T}_{\text{bin}})\big),$$

where LP_Q, LP_K and LP_V are separate learned linear projections for queries, keys, and values.

To capture a refined global structure, we update the bin tokens based on aggregated pixel-to-bin responses. In the second pass **(bin-to-pixel)**, the roles are reversed. We first compute updated bin tokens $\mathbf{T}'_{\text{bin}} \in \mathbb{R}^{M\times C}$ derived from $\mathbf{F}_{\text{pix}\to\text{bin}}$. These updated tokens encode refined depth-bin representations, allowing them to more accurately modulate the per-pixel features. These tokens then query the pixel fea-

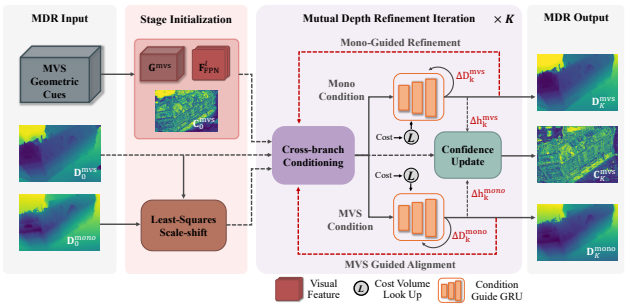


Figure 5. Architecture of the mutual depth refinement module. The module employs two symmetric recurrent (GRU) branches that iteratively update the MVS and monocular depth maps along with the MVS confidence map.

tures through cross-attention, allowing each pixel to receive refined global information from its corresponding depth bin:

$$\mathbf{F}_{\text{bin}\to\text{pix}} = \text{Attn}\big(\text{LP}_Q(\mathbf{T}'_{\text{bin}}),\, \text{LP}_K(\mathbf{F}_{\text{flatten}}),\, \text{LP}_V(\mathbf{F}_{\text{flatten}})\big).$$

We then aggregate the bin-to-pixel attention outputs back into the pixel space using the transposed attention weights, $\mathbf{W}^T_{\text{bin}\to\text{pix}}$, which project the global responses from the bin domain to their corresponding pixel locations:

$$\tilde{\mathbf{F}}_{\text{bin}\to\text{pix}} = \mathbf{W}^T_{\text{bin}\to\text{pix}}\mathbf{F}_{\text{bin}\to\text{pix}}.$$

This result is combined with the original pixel features via a residual connection and LayerNorm to produce the final attention-enhanced features:

$$\mathbf{F}_{\text{attn}} = \text{LayerNorm}\big(\mathbf{F}_{\text{flatten}} + \tilde{\mathbf{F}}_{\text{bin}\to\text{pix}}\big) \in \mathbb{R}^{S\times C}.$$

Finally, the enhanced features $\mathbf{F}_{\text{attn}}$ are reshaped back to their 3D volume structure and processed sequentially through a 1×1×1 convolution, 3D LayerNorm, and a softmax activation. This produces the probability volume $\mathbf{PV} \in [0,1]^{D\times H_1\times W_1}$.

3.2. Mutual Depth Refinement

At each stage, we perform mutual refinement between the monocular depth $\mathbf{D}^{\text{mono}}$ and the MVS depth $\mathbf{D}^{\text{mvs}}$. This process corrects fine-grained scale inconsistencies of $\mathbf{D}^{\text{mono}}$ at the pixel level and enhances the structural fidelity of $\mathbf{D}^{\text{mvs}}$. Unlike prior work [21] applying only a global scale-shift, our approach iteratively refines pixel-level residuals, yielding locally accurate scale alignment.

While conceptually related to mutual refinement strategies in two-view stereo [9, 36], these methods are not directly applicable to MVS since they do not fully account for multi-view coupling and confidence modeling. We therefore propose an architecture specifically designed for robust and consistent monocular scale alignment in multi-view settings.

As illustrated in Fig. 5, our framework consists of two symmetric branches: **MVS-Guided Alignment** (MVS-GA) and **Mono-Guided Refinement** (Mono-GR). These branches operate iteratively, with a shared ConvGRU [11] architecture predicting residual updates to refine each other's depth estimates at each iteration. This design effectively resolves global scale inconsistencies with conservative, fine-grained updates preserving detailed geometry.

MVS-Guided Alignment (MVS-GA). The MVS-GA branch aligns the pixel-level scale of the monocular depth by leveraging geometric cues from the MVS cost volumes. We first apply a global least-squares scale-shift to $\mathbf{D}^{\text{mono}}$ for coarse matching with $\mathbf{D}^{\text{mvs}}$. Next, we construct a geometric feature volume $\mathbf{G}^{\text{mvs}}$ by concatenating three components: (i) the raw cost volume $\mathbf{CV}_{\text{multi}}$, (ii) the probability volume $\mathbf{PV}$, and (iii) a curvature volume $\overline{\mathbf{PV}}$. The curvature volume $\overline{\mathbf{PV}}$ is computed as the second-order variation of $\mathbf{PV}$ along depth, quantifying peak sharpness per pixel and serving as a confidence proxy near depth boundaries and in unambiguous regions. The resulting feature volume $\mathbf{G}^{\text{mvs}} = [\mathbf{CV}_{\text{multi}}, \mathbf{PV}, \overline{\mathbf{PV}}]$ thus encodes strong MVS constraints for refinement.

At each refinement step k, the module constructs a conditioning feature tensor $\mathbf{x}^{\text{mvs}}_k$, which concatenates the encoded representations of the MVS geometry ($\mathbf{G}^{\text{mvs}}$), the current MVS depth ($\mathbf{D}^{\text{mvs}}_k$) and confidence ($\mathbf{C}^{\text{mvs}}_k$), and the current monocular depth ($\mathbf{D}^{\text{mono}}_k$).

$$\begin{aligned}\mathbf{x}^{\text{mvs}}_k = [&\text{Enc}_g(\mathbf{G}^{\text{mvs}}), \text{Enc}_{\text{mvs}}(\mathbf{D}^{\text{mvs}}_k, \mathbf{C}^{\text{mvs}}_k),\\ &\text{Enc}_{\text{mono}}(\mathbf{D}^{\text{mono}}_k), \mathbf{D}^{\text{mono}}_k],\end{aligned}$$

where Enc_g, Enc_{mvs}, and Enc_{mono} are lightweight two-layer convolutional encoders [9]. For the initial step ($k=0$), $\mathbf{D}^{\text{mvs}}_0$ and $\mathbf{D}^{\text{mono}}_0$ are initialized with the initial depth maps, and $\mathbf{C}^{\text{mvs}}_0$ is the confidence map estimated from $\mathbf{PV}$ [7].

The conditioning feature $\mathbf{x}^{\text{mvs}}_k$ is then fed into a ConvGRU, together with the previous hidden state $\mathbf{h}^{\text{mono}}_{k-1}$, to update the monocular hidden state:

$$\begin{aligned}\mathbf{z}_k &= \sigma(\text{Conv}([\mathbf{h}^{\text{mono}}_{k-1}, \mathbf{x}^{\text{mvs}}_k], \mathbf{W}_z) + \mathbf{c}_z),\\ \mathbf{r}_k &= \sigma(\text{Conv}([\mathbf{h}^{\text{mono}}_{k-1}, \mathbf{x}^{\text{mvs}}_k], \mathbf{W}_r) + \mathbf{c}_r),\\ \hat{\mathbf{h}}^{\text{mono}}_k &= \tanh(\text{Conv}([\mathbf{r}_k \odot \mathbf{h}^{\text{mono}}_{k-1}, \mathbf{x}^{\text{mvs}}_k], \mathbf{W}_h) + \mathbf{c}_h),\\ \mathbf{h}^{\text{mono}}_k &= (1-\mathbf{z}_k)\odot \mathbf{h}^{\text{mono}}_{k-1} + \mathbf{z}_k \odot \hat{\mathbf{h}}^{\text{mono}}_k,\end{aligned}$$

where $\mathbf{W}_*$ and $\mathbf{c}_*$ denote learned convolutional weights and biases, and $\odot$ represents element-wise multiplication. Finally, a lightweight prediction head f_θ regresses the residual depth $\Delta\mathbf{D}^{\text{mono}}_k$ from the updated hidden state $\mathbf{h}^{\text{mono}}_k$, and the monocular depth is updated as:

$$\Delta\mathbf{D}^{\text{mono}}_k = f_\theta(\mathbf{h}^{\text{mono}}_k), \quad \mathbf{D}^{\text{mono}}_{k+1} = \mathbf{D}^{\text{mono}}_k + \Delta\mathbf{D}^{\text{mono}}_k.$$

Mono-Guided Refinement (Mono-GR). The Mono-GR branch refines the MVS depth by incorporating structural priors from the scale-aligned monocular depth. This process is particularly effective in regions with low texture or occlusion, where photometric consistency alone is unreliable.

Complementary to the MVS-GA branch, Mono-GR employs a symmetric ConvGRU network that leverages monocular cues to enhance structural completeness. At each iteration k, it constructs a conditioning feature $\mathbf{x}_k^{\text{mono}}$ by aggregating multiple inputs:

$$\mathbf{x}_k^{\text{mono}} = [\text{Enc}_{\text{mono}}(\mathbf{D}_k^{\text{mono}}), \text{Enc}_{\text{cross}}(\mathbf{D}_k^{\text{mono}}, \mathbf{D}_k^{\text{mvs}}, \mathbf{C}_k^{\text{mvs}}), \mathbf{F}_{\text{FPN}}^{l}, \text{Enc}_{\text{mvs}}(\mathbf{D}_k^{\text{mvs}})],$$

where $\text{Enc}_{\text{cross}}$ is also a lightweight CNN encoder, which enforces cross-modal consistency between $\mathbf{D}_k^{\text{mono}}$ and $\mathbf{D}_k^{\text{mvs}}$, weighted by $\mathbf{C}_k^{\text{mvs}}$.

The ConvGRU updates its hidden state $\mathbf{h}_k^{\text{mvs}}$ analogously to the MVS-GA branch, and a two-layer CNN prediction head f_ϕ regresses the residual update for the MVS depth:

$$\Delta\mathbf{D}_k^{\text{mvs}} = f_\phi(\mathbf{h}_k^{\text{mvs}}), \quad \mathbf{D}_{k+1}^{\text{mvs}} = \mathbf{D}_k^{\text{mvs}} + \Delta\mathbf{D}_k^{\text{mvs}}.$$

In parallel, a two-layer CNN g_ψ predicts an updated confidence map $\hat{\mathbf{C}}_k^{\text{mvs}}$ by fusing the hidden features from both branches:

$$\hat{\mathbf{C}}_k^{\text{mvs}} = g_\psi([\mathbf{h}_k^{\text{mvs}}, \mathbf{h}_k^{\text{mono}}]).$$

To ensure temporal stability and suppress abrupt confidence changes across iterations, we update the confidence map using an exponential moving average with $\alpha \in (0, 1)$:

$$\mathbf{C}_{k+1}^{\text{mvs}} = (1-\alpha)\cdot\mathbf{C}_k^{\text{mvs}} + \alpha\cdot\hat{\mathbf{C}}_k^{\text{mvs}}.$$

This strategy mitigates abrupt confidence fluctuations across iterations, ensuring stable training and inference. The consistently updated confidence map $\mathbf{C}_{k+1}^{\text{mvs}}$ is then used in the subsequent iteration to weight the cross-modal consistency loss, which prioritizes refinement in uncertain regions.

After completing the iterative dual-branch refinement, the scale-corrected monocular depth $\mathbf{D}_K^{\text{mono}}$ and structurally enhanced MVS depth $\mathbf{D}_K^{\text{mvs}}$ are forwarded to the next cascade stage. In this stage, $\mathbf{D}_K^{\text{mvs}}$ serves as the basis for generating depth hypotheses, while $\mathbf{D}_K^{\text{mono}}$ is upscaled and reused for its subsequent mutual refinement.

3.3. Loss Function

We train the cascade MVS framework using three complementary losses applied at every stage l. First, we adopt the standard cross-entropy loss $\mathcal{L}_{\text{CE}}^{(l)}$, to measure depth accuracy at stage l, i.e., the divergence between the predicted and ground-truth depth distributions [29].

Second, we introduce a mutual refinement loss $\mathcal{L}_{\text{mut}}^{(l)}$. Because each stage outputs K intermediate MVS depth maps $\{\mathbf{D}_k^{\text{mvs},(l)}\}_{k=1}^{K}$, we supervise all of them with a decayed ℓ_1 loss ($r = 0.9$) to emphasize later refinements [26]:

$$\mathcal{L}_{\text{mut}}^{(l)} = \frac{1}{K}\sum_{k=1}^{K} r^{(K-k)} \|\mathbf{D}_k^{\text{mvs},(l)} - \mathbf{D}^{\text{gt},(l)}\|_1.$$

Finally, we add an order-preserving loss $\mathcal{L}_{\text{ord}}^{(l)}$ for the MVS depth map of each stage l. Inspired by [26], to maintain local structure and correct depth ordering, we sample N_{pair} pixel pairs (p, q) and penalize order violations [8]:

$$\mathcal{L}_{\text{ord}}^{(l)} = \frac{1}{N_{\text{pair}}}\sum_{(p,q)} \max\Big(0,\ \big(\mathbf{D}^{\text{mvs},(l)}(p) - \mathbf{D}^{\text{mvs},(l)}(q)\big) \cdot \text{sign}\big(\mathbf{D}^{\text{gt},(l)}(q) - \mathbf{D}^{\text{gt},(l)}(p)\big)\Big),$$

where (p, q) are randomly sampled pixel pairs from valid depth regions. The $\text{sign}(\cdot)$ function returns -1 or $+1$ depending on the relative depth ordering between the two pixels. The loss is zero when the predicted ordering matches the ground truth and increases linearly otherwise.

The final loss $\mathcal{L}_{\text{total}}$ aggregates all stages:

$$\mathcal{L}_{\text{total}} = \sum_{l=1}^{4}\big(\mathcal{L}_{\text{CE}}^{(l)} + \lambda_{\text{mut}}\,\mathcal{L}_{\text{mut}}^{(l)} + \lambda_{\text{ord}}\,\mathcal{L}_{\text{ord}}^{(l)}\big).$$

To balance with the main cross-entropy loss, we set the weights $\lambda_{\text{mut}} = \lambda_{\text{ord}} = 0.02$.

4. Experiment

We evaluate the 3D reconstruction performance of our method on the DTU [1] and Tanks and Temples (TNT) [23] benchmarks. To examine the generalizability of our depth estimation, we also test on the RobustMVD benchmark [34], which offers ground-truth depth maps for DTU, TNT, and KITTI [15]. In addition, we conduct some ablation studies.

4.1. Implementation Details

Hyperparameters. Depth estimation utilizes a four-stage coarse-to-fine pipeline with inverse depth sampling of (8, 8, 4, 4) hypotheses per stage and corresponding feature dimensionalities (64, 32, 16, 8). The number of depth bin masks M is 10, and the refinement iterations K per stage are 12, 8, 5, and 3, respectively.

Training. For fair comparison, we train and evaluate on the DTU dataset [1], adhering to standard data splits and view selection protocols [7, 42]. The model is trained using 5 input views with a resolution of 512×640. After DTU training, the model is fine-tuned on the BlendedMVS dataset [43] using 11 input views with a resolution of 576×768.

Testing. We fuse depth maps using a dynamic fusion strategy [38]. For DTU evaluation, we perform inference at 1152×1536 resolution, and the fused point clouds are used for quantitative evaluation. For the TNT benchmark [23], we follow [7], performing inference at 1920×1088 and applying dynamic fusion to generate the final 3D point clouds. On the RobustMVD benchmark [34], our pre-trained model is evaluated directly without fine-tuning. All experiments were conducted on a workstation with an Intel i9-13900KS processor and an NVIDIA GeForce RTX 3090 Ti GPU.

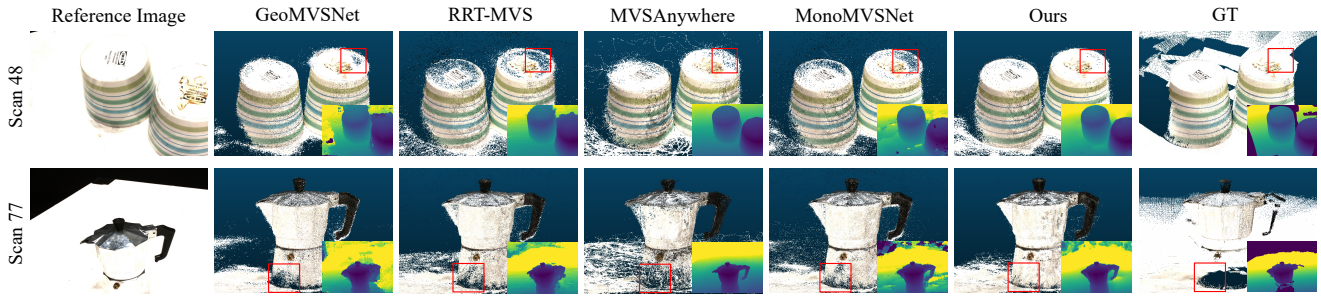


Figure 6. Qualitative results on the DTU dataset. Our method outperforms state-of-the-art approaches, producing highly complete 3D reconstructions with clean depth maps even on the challenging Scans 48 and 77.

Method	**Acc.** $(mm)\downarrow$	**Comp.** $(mm)\downarrow$	**Overall** $(mm)\downarrow$
Gipuma [14]	**0.283**	0.873	0.578
COLMAP [33]	0.400	0.664	0.532
CDS-MVSNet [16]	0.351	0.278	0.315
RA-MVSNet [45]	0.326	0.268	0.297
GeoMVSNet [46]	0.331	0.259	0.295
ET-MVSNet [27]	0.329	0.253	0.291
TransMVSNet [12]	0.321	0.289	0.305
WT-MVSNet [24]	0.309	0.281	0.295
MVSFormer [6]	0.327	0.251	0.289
MVSFormer++ [7]	0.309	0.252	0.281
GoMVS [37]	0.347	**0.227**	0.287
MVSAnywhere [18]	0.845	0.625	0.735
RRT-MVS [22]	0.309	0.261	0.285
MonoMVSNet [21]	0.313	0.243	0.278
M2Depth(Ours)	0.301	0.248	**0.274**

Table 1. Quantitative results on the DTU evaluation dataset using distance metrics $[mm]$. Lower is better for all metrics.

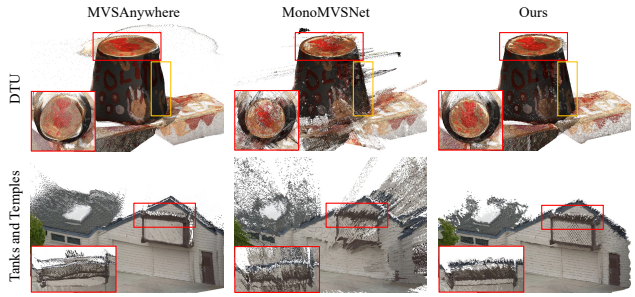


Figure 7. Qualitative comparison of two-view reconstruction. Compared to MonoMVSNet, which produces significant noise, and MVSAnywhere, which suffers from scale ambiguity, our method generates complete and well-aligned 3D surfaces.

4.2. Benchmark Performance

Evaluation on DTU Dataset. Table 1 presents the quantitative results on the DTU evaluation set, based on standard point cloud reconstruction metrics. Our method significantly outperforms all prior state-of-the-art approaches in terms of overall error. Leveraging guidance from the DFM, our approach generates more complete depth maps, resulting in notably better completeness. Fig. 6 illustrates qualitative results, showing that our method reconstructs finer and more detailed point clouds, particularly in challenging regions. Notably, on scans 48 and 77, widely considered the most difficult cases, our method produces robust and accurate depth estimations, leading to complete 3D reconstructions.

Evaluation on TNT. To assess the generalization capability of our method, we evaluate its reconstruction performance using F-scores on the TNT benchmark. Table 2 presents the quantitative results for both the intermediate and advanced sets. Our method outperforms existing approaches across both sets, demonstrating its strong generalization ability.

Evaluation on RobustMVD. To assess the generalization of our method in depth estimation, we evaluate it on the RobustMVD benchmark. We compare our method against DFMs (Depth Pro [5], DepthAnythingV2 [41]), MVS methods (PatchmatchNet[35] and MVSFormer++[7]), and the hybrid approaches (MonoMVSNet [21] and MVSAnywhere [18]). Following [34], Table 3 reports two metrics comparing the predicted depth $\hat{d}$ and ground-truth depth d: (1) Absolute Relative Error (rel), computed as $|\hat{d} - d|/d$ per pixel, and (2) Inlier Ratio (τ), the percentage of pixels where $\max\left(\frac{d}{\hat{d}}, \frac{\hat{d}}{d}\right) < 1.03$.

Although DFM-based depth estimates are aligned to the ground-truth scale via post-processing, they still exhibit the lowest depth estimation performance. MVS methods perform well on the DTU dataset, on which they were trained, but their performance drops significantly on other datasets. Similarly, MonoMVSNet has a high dependency on MVS; consequently, even when leveraging DFM, its performance does not significantly differ from that of MVS. In contrast, MVSAnywhere focuses more on DFM, which results in better generalization in depth estimation.

Our method achieves the best results on both the DTU and TNT benchmarks. On the KITTI dataset, it performs slightly worse than MVSAnywhere due to challenges with dynamic objects, though the gap is minimal. In summary, while pure MVS methods and MVS-centric hybrids (MonoMVSNet) excel at 3D reconstruction but falter in depth estimation, and

Methods	Intermediate									Advanced						
	Mean	Fam.	Fra.	Hor.	Lig.	M60	Pan.	Pla.	Tra.	Mean	Aud.	Bal.	Cou.	Mus.	Pal.	Tem.
COLMAP [33]	42.14	50.41	22.25	25.63	56.43	44.83	46.97	48.53	42.04	27.24	16.02	25.23	34.70	41.51	18.05	27.94
CasMVSNet[17]	56.84	76.37	58.45	46.26	55.81	56.11	54.06	58.18	49.51	31.12	19.81	38.46	29.10	43.87	27.36	28.11
TransMVSNet[12]	63.52	80.92	65.83	56.94	62.54	63.06	60.00	60.20	58.67	37.00	24.84	44.59	34.77	46.49	34.69	36.62
WT-MVSNet[24]	65.34	81.87	67.33	57.76	64.77	65.68	64.61	62.35	58.38	39.91	29.20	44.48	39.55	53.49	34.57	38.15
RA-MVSNet[45]	65.72	82.44	66.61	58.40	64.78	**67.14**	65.60	62.74	58.08	39.93	29.14	46.04	40.30	53.22	34.63	36.28
DMVSNet[44]	64.66	81.27	67.54	59.10	63.12	64.64	64.80	59.83	56.97	41.17	30.08	46.10	40.65	53.53	35.08	41.60
MVSFormer[6]	66.37	82.06	69.34	60.49	68.61	65.67	64.08	61.23	59.53	40.87	28.22	46.75	39.30	52.88	35.16	42.95
MVSFormer++[7]	67.18	**82.69**	69.44	64.24	69.16	64.13	66.43	61.19	60.12	41.60	29.93	45.69	39.46	53.58	35.56	45.39
RRT-MVS[22]	68.16	82.54	72.31	61.44	69.89	65.32	68.88	64.45	60.48	43.29	30.95	46.42	41.13	55.46	**37.63**	48.12
GoMVS[37]	66.44	82.68	69.23	**69.19**	63.56	65.13	62.10	58.81	60.80	43.07	**35.52**	47.15	42.52	52.08	36.34	44.82
MonoMVSNet[21]	68.63	82.38	**72.89**	62.80	70.49	65.79	68.54	65.54	60.59	43.58	30.33	46.76	42.90	**56.31**	37.28	47.88
M2Depth(Ours)	**68.87**	82.55	72.70	62.46	**70.65**	65.29	**70.22**	**65.58**	**61.55**	**43.85**	31.36	**47.26**	**42.91**	55.69	37.61	**48.26**

Table 2. Quantitative F-score results on the TNT benchmark. Our method achieves the highest mean F-scores on both the intermediate and advanced sets.

Model	DTU		TNT		KITTI	
	rel ↓	τ ↑	rel ↓	τ ↑	rel ↓	τ ↑
Depth Pro [5]	5.6	49.6	5.6	57.5	6.1	39.6
DepthAnythingV2 [41]	2.6	74.7	4.5	57.5	6.6	38.6
PatchmatchNet [35]	2.1	82.8	4.8	82.9	10.8	45.8
MVSFormer++ [7]	0.9	95.3	3.2	88.1	4.4	65.7
MonoMVSNet [21]	0.9	95.5	2.6	89.0	4.1	66.2
MVSAnywhere [18]	1.3	95.0	2.1	90.5	**3.2**	**68.8**
M2Depth(Ours)	**0.8**	**96.2**	**2.0**	**90.7**	3.7	66.8

Table 3. Quantitative depth results on the RobustMVD benchmark, reported using Absolute Relative Error (rel) and Inlier Ratio (τ).

Model	CC		MDR	Loss		Overall↓	Acc.↓	Comp.↓	MAE↓	Mem↓	Time↓
	MV	BM		Mut	Ord						
A						0.292	0.320	0.264	6.01	2.12	0.22
B	✓					0.288	0.314	0.262	5.82	2.52	0.25
C		✓				0.289	0.317	0.261	5.86	2.46	0.23
D	✓	✓				0.285	0.311	0.257	5.44	2.77	0.26
E	✓	✓	✓			0.277	0.304	0.250	4.91	3.12	0.50
F	✓	✓	✓	✓		0.275	0.293	0.257	4.88	3.12	0.50
G	✓	✓	✓	✓	✓	0.274	0.301	0.248	4.84	3.12	0.50

Table 4. Ablation study on the components of our method on the DTU dataset: Cost-Crafter (CC), Monocular Volume Fusion (MV), Bin Mask-based Refinement (BM), Mutual Depth Refinement (MDR), Mutual Refinement Loss (Mut), and Order-preserving Loss (Ord). We also report Mean Absolute Error (MAE) in mm, and average GPU memory (GB) and runtime (seconds).

DFM-centric hybrids (MVSAnywhere) exhibit the opposite pattern, our method excels in both tasks, achieving state-of-the-art performance overall.

4.3. Ablation Study

Model Ablation. We performed an ablation study on the DTU dataset to evaluate the contribution of each key component in our method. Table 4 summarizes the performance of different model variants. As components are incrementally added from the baseline (Model-A) to our full method (Model-G), both the overall reconstruction error and depth error progressively decrease. Notably, Model-D, which incorporates the full Cost-Crafter module, shows a substantial performance gain over the baseline. Furthermore, Model-E, which introduces the mutual depth refinement module, achieves the single largest improvement. The final loss functions provide additional incremental refinements, demonstrating that each component plays a complementary role in achieving our final performance at the cost of increased memory and runtime.

Sparse View Reconstruction. To evaluate robustness in challenging scenarios with occlusions or limited view overlap, we assessed MVS performance using only two views. Fig. 1 and Fig. 7 illustrate the two-view reconstruction results from DFM-based MVS approaches. The results show MonoMVSNet produced significant noise and artifacts in occluded or non-overlapping regions. MVSAnywhere generated cleaner models overall, but still suffered from scale ambiguity. This resulted in misaligned surfaces. In contrast, our method successfully reconstructed regions even without view overlap, producing well-aligned, scale-corrected 3D surfaces. This demonstrates our method effectively integrates the strengths of both DFM and MVS, achieving robust reconstructions in sparse-view conditions.

5. Conclusion

We introduced a novel framework unifying monocular depth priors from DFMs with a cascade MVS pipeline. Unlike existing methods that rely on static, one-way fusion, we introduce two key contributions: (i) a prior-guided cost volume refinement to establish a robust geometric foundation, and (ii) a bidirectional mutual refinement strategy that iteratively corrects the monocular prior's scale while enhancing MVS completeness. Extensive experiments on the DTU, TNT, and RobustMVD benchmarks show our method surpasses state-of-the-art approaches. Our framework resolves the trade-off of prior hybrid methods [18, 21], excelling in both complete 3D reconstruction and generalizable depth estimation.

References

[1] Henrik Aanæs, Rasmus Ramsbøl Jensen, George Vogiatzis, Engin Tola, and Anders Bjorholm Dahl. Large-scale data for multiple-view stereopsis. *International Journal of Computer Vision*, pages 1–16, 2016. 2, 6

[2] Muhammad Awais, Muzammal Naseer, Salman Khan, Rao Muhammad Anwer, Hisham Cholakkal, Mubarak Shah, Ming-Hsuan Yang, and Fahad Shahbaz Khan. Foundation models defining a new era in vision: a survey and outlook. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025. 1

[3] Luca Bartolomei, Fabio Tosi, Matteo Poggi, and Stefano Mattoccia. Stereo anywhere: Robust zero-shot deep stereo matching even where either stereo or mono fail. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 1013–1027, 2025. 2, 3

[4] Michael Bleyer, Christoph Rhemann, and Carsten Rother. Patchmatch stereo-stereo matching with slanted support windows. In *Bmvc*, pages 1–11, 2011. 1

[5] Aleksei Bochkovskii, AmaÃĢl 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. 7, 8

[6] Chenjie Cao, Xinlin Ren, and Yanwei Fu. Mvsformer: Multi-view stereo by learning robust image features and temperature-based depth. *arXiv preprint arXiv:2208.02541*, 2022. 2, 7, 8

[7] Chenjie Cao, Xinlin Ren, and Yanwei Fu. Mvsformer++: Revealing the devil in transformer's details for multi-view stereo. *arXiv preprint arXiv:2401.11673*, 2024. 2, 3, 5, 6, 7, 8

[8] Weifeng Chen, Zhao Fu, Dawei Yang, and Jia Deng. Single-image depth perception in the wild. *Advances in neural information processing systems*, 29, 2016. 6

[9] Junda Cheng, Longliang Liu, Gangwei Xu, Xianqi Wang, Zhaoxing Zhang, Yong Deng, Jinliang Zang, Yurui Chen, Zhipeng Cai, and Xin Yang. Monster: Marry monodepth to stereo unleashes power. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 6273–6282, 2025. 3, 5

[10] Shuo Cheng, Zexiang Xu, Shilin Zhu, Zhuwen Li, Li Erran Li, Ravi Ramamoorthi, and Hao Su. Deep stereo using adaptive thin volume representation with uncertainty awareness. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2524–2534, 2020. 2

[11] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014. 5

[12] Yikang Ding, Wentao Yuan, Qingtian Zhu, Haotian Zhang, Xiangyue Liu, Yuanjiang Wang, and Xiao Liu. Transmvsnet: Global context-aware multi-view stereo network with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8585–8594, 2022. 2, 7, 8

[13] Silvano Galliani, Katrin Lasinger, and Konrad Schindler. Massively parallel multiview stereopsis by surface normal diffusion. In *Proceedings of the IEEE international conference on computer vision*, pages 873–881, 2015. 1

[14] Silvano Galliani, Katrin Lasinger, and Konrad Schindler. Gipuma: Massively parallel multi-view stereo reconstruction. *Publikationen der Deutschen Gesellschaft für Photogrammetrie, Fernerkundung und Geoinformation e. V*, 25(361-369):2, 2016. 7

[15] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012. 6

[16] Khang Truong Giang, Soohwan Song, and Sungho Jo. Curvature-guided dynamic scale networks for multi-view stereo. *arXiv preprint arXiv:2112.05999*, 2021. 7

[17] Xiaodong Gu, Zhiwen Fan, Siyu Zhu, Zuozhuo Dai, Feitong Tan, and Ping Tan. Cascade cost volume for high-resolution multi-view stereo and stereo matching. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2495–2504, 2020. 1, 2, 3, 8

[18] Sergio Izquierdo, Mohamed Sayed, Michael Firman, Guillermo Garcia-Hernando, Daniyar Turmukhambetov, Javier Civera, Oisin Mac Aodha, Gabriel Brostow, and Jamie Watson. Mvsanywhere: Zero-shot multi-view stereo. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 11493–11504, 2025. 1, 2, 3, 4, 7, 8

[19] Mengqi Ji, Juergen Gall, Haitian Zheng, Yebin Liu, and Lu Fang. Surfacenet: An end-to-end 3d neural network for multiview stereopsis. In *Proceedings of the IEEE international conference on computer vision*, pages 2307–2315, 2017. 1, 2

[20] Hualie Jiang, Zhiqiang Lou, Laiyan Ding, Rui Xu, Minglang Tan, Wenjie Jiang, and Rui Huang. Defom-stereo: Depth foundation model based stereo matching. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 21857–21867, 2025. 2, 3

[21] Jianfei Jiang, Qiankun Liu, Haochen Yu, Hongyuan Liu, Liyong Wang, Jiansheng Chen, and Huimin Ma. Monomvsnet: Monocular priors guided multi-view stereo network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 27806–27816, 2025. 1, 2, 3, 4, 5, 7, 8

[22] Jianfei Jiang, Liyong Wang, Haochen Yu, Tianyu Hu, Jiansheng Chen, and Huimin Ma. Rrt-mvs: Recurrent regularization transformer for multi-view stereo. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3994–4002, 2025. 7, 8

[23] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics*, 36(4), 2017. 2, 6

[24] Jinli Liao, Yikang Ding, Yoli Shavit, Dihe Huang, Shihao Ren, Jia Guo, Wensen Feng, and Kai Zhang. Wt-mvsnet: window-based transformers for multi-view stereo. *Advances in Neural Information Processing Systems*, 35:8564–8576, 2022. 2, 7, 8

[25] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid

networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. 3

[26] Lahav Lipson, Zachary Teed, and Jia Deng. Raft-stereo: Multilevel recurrent field transforms for stereo matching. In *2021 International Conference on 3D Vision (3DV)*, pages 218–227. IEEE, 2021. 6

[27] Tianqi Liu, Xinyi Ye, Weiyue Zhao, Zhiyu Pan, Min Shi, and Zhiguo Cao. When epipolar constraint meets non-local operators in multi-view stereo. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 18088–18097, 2023. 7

[28] Keyang Luo, Tao Guan, Lili Ju, Yuesong Wang, Zhuo Chen, and Yawei Luo. Attention-aware multi-view stereo. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1590–1599, 2020. 2

[29] Anqi Mao, Mehryar Mohri, and Yutao Zhong. Cross-entropy loss functions: Theoretical analysis and applications. In *International conference on Machine learning*, pages 23803–23828. pmlr, 2023. 6

[30] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 1

[31] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021. 1

[32] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 12179–12188, 2021. 2

[33] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 7, 8

[34] Philipp Schröppel, Jan Bechtold, Artemij Amiranashvili, and Thomas Brox. A benchmark and a baseline for robust multi-view depth estimation. In *Proceedings of the International Conference on 3D Vision (3DV)*, 2022. 2, 6, 7

[35] Fangjinhua Wang, Silvano Galliani, Christoph Vogel, Pablo Speciale, and Marc Pollefeys. Patchmatchnet: Learned multi-view patchmatch stereo. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14194–14203, 2021. 7, 8

[36] Bowen Wen, Matthew Trepte, Joseph Aribido, Jan Kautz, Orazio Gallo, and Stan Birchfield. Foundationstereo: Zero-shot stereo matching. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 5249–5260, 2025. 2, 3, 5

[37] Jiang Wu, Rui Li, Haofei Xu, Wenxun Zhao, Yu Zhu, Jinqiu Sun, and Yanning Zhang. Gomvs: Geometrically consistent cost aggregation for multi-view stereo. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20207–20216, 2024. 2, 7, 8

[38] Jianfeng Yan, Zizhuang Wei, Hongwei Yi, Mingyu Ding, Runze Zhang, Yisong Chen, Guoping Wang, and Yu-Wing Tai. Dense hybrid recurrent multi-view stereo net with dynamic consistency checking. In *European conference on computer vision*, pages 674–689. Springer, 2020. 6

[39] Jiayu Yang, Wei Mao, Jose M Alvarez, and Miaomiao Liu. Cost volume pyramid based depth inference for multi-view stereo. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4877–4886, 2020. 2

[40] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10371–10381, 2024. 2, 3

[41] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *Advances in Neural Information Processing Systems*, 37: 21875–21911, 2024. 1, 2, 3, 7, 8

[42] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. In *Proceedings of the European conference on computer vision (ECCV)*, pages 767–783, 2018. 1, 2, 6

[43] Yao Yao, Zixin Luo, Shiwei Li, Jingyang Zhang, Yufan Ren, Lei Zhou, Tian Fang, and Long Quan. Blendedmvs: A large-scale dataset for generalized multi-view stereo networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1790–1799, 2020. 6

[44] Xinyi Ye, Weiyue Zhao, Tianqi Liu, Zihao Huang, Zhiguo Cao, and Xin Li. Constraining depth map geometry for multi-view stereo: A dual-depth approach with saddle-shaped depth cells. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17661–17670, 2023. 8

[45] Yisu Zhang, Jianke Zhu, and Lixiang Lin. Multi-view stereo representation revist: Region-aware mvsnet. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 17376–17385, 2023. 7, 8

[46] Zhe Zhang, Rui Peng, Yuxi Hu, and Ronggang Wang. Geomvsnet: Learning multi-view stereo with geometry perception. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 21508–21518, 2023. 2, 7

[47] Jingyi Zhou, Haoyu Zhang, Jiakang Yuan, Peng Ye, Tao Chen, Hao Jiang, Meiya Chen, and Yangyang Zhang. All-in-one: Transferring vision foundation models into stereo matching. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 10797–10805, 2025. 2, 3