IVGT: IMPLICIT VISUAL GEOMETRY TRANSFORMER FOR NEURAL SCENE REPRESENTATION

Anonymous authors
Paper under double-blind review

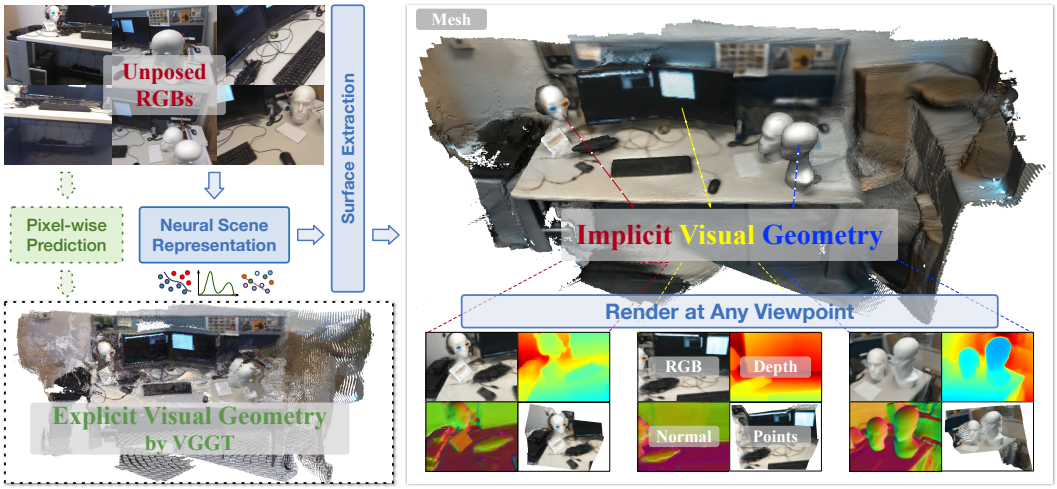

Figure 1: **IVGT** implicitly models coherent 3D geometry and appearance from pose-free multi-view images in one feedforward pass. It learns a continuous neural scene representation in a global canonical coordinate system, enabling various downstream tasks including mesh reconstruction, point cloud reconstruction, novel-view synthesis, and surface estimation across diverse scenes.

ABSTRACT

Reconstructing coherent 3D geometry and appearance from unposed multi-view images is a fundamental yet challenging problem in computer vision. Most existing visual geometry foundation models predict explicit geometry by regressing pixel-aligned pointmaps, often suffering from redundancy and limited geometric continuity. We propose **IVGT**, an **I**mplicit **V**isual **G**eometry **T**ransformer that implicitly models continuous and coherent geometry from pose-free multi-view images. This formulation learns a continuous neural scene representation in a canonical coordinate system and supports continuous spatial queries at any 3D positions, retrieving local features to predict signed distance (SDF) values and colors using lightweight decoders. It allows direct extraction of continuous and coherent surface geometry, enabling rendering of RGB images, depth maps, and surface normal maps from arbitrary viewpoints. We train IVGT via multi-dataset joint optimization with 2D supervision and 3D geometric regularization. IVGT demonstrates generalization across scenes and achieves strong performance on various tasks, including mesh and point cloud reconstruction, novel view synthesis, depth and surface normal estimation, and camera pose estimation.

1 INTRODUCTION

Reconstructing 3D scenes from multi-view images (Snavely et al., 2006; Furukawa & Ponce, 2009; Schönberger & Frahm, 2016) has long been a central problem in computer vision, with wide applications in robotics (Wang et al., 2024b; Wu et al., 2025b) and autonomous driving (Huang et al., 2023; Zuo et al., 2026). Given multi-view images, the goal is to recover a representation that captures both accurate geometry and view-consistent appearance, while supporting flexible rendering and downstream reasoning (Mildenhall et al., 2021; Yu et al., 2021).

The recent emergence of visual geometry foundation models (Wang et al., 2024a; 2025a) has significantly revolutionized the paradigm of 3D reconstruction. By adopting an end-to-end formulation and

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

large-scale pretraining, they directly regress multi-view observations into a shared coordinate system without requiring explicit camera poses (Wang & Agapito, 2024; Wang et al., 2025b; Wu et al., 2025a; Zhuo et al., 2026; Chen et al., 2025; Lan et al., 2025; Fang et al., 2025). They decode pixel-aligned features into explicit 3D point coordinates, simplifying the conventional complex reconstruction pipeline. However, this explicit pixel-aligned formulation usually leads to redundant representations and lacks geometric continuity. Learning a geometry foundation model that supports coherent surface reconstruction and continuous rendering is of great value but generally remains unexplored.

To bridge this gap, we propose an **I**mplicit **V**isual **G**eometry **T**ransformer (**IVGT**) that learns a global neural scene representation from pose-free multi-view images to implicitly model continuous and coherent geometry, as shown in Figure 1. Specifically, a transformer backbone jointly processes tokens from all input views through alternating intra-view and cross-view attention, aggregating multi-view observations into a unified canonical representation without requiring explicit camera poses as input. This representation implicitly aligns multiple views within a shared coordinate system and supports continuous spatial queries over the 3D space. Given an arbitrary 3D query point, the model projects it onto each input view to retrieve the corresponding pixel-aligned features, which are further augmented with view-relative depth encodings to disambiguate points lying along the same projection ray. The aggregated feature is then decoded by cascaded MLP decoders: a geometry decoder to predict the signed distance function (SDF) value at the query point, and an appearance decoder to take the SDF gradient (i.e., the surface normal) and the viewing direction to predict view-dependent color. This implicit SDF-based formulation enables unified modeling of geometry and appearance under a single continuous representation. The predicted SDF values are converted to densities to support differentiable volume rendering, allowing the model to render RGB images, depth maps, and surface normal maps from arbitrary novel viewpoints.

To fully leverage large-scale datasets, we design a multi-stage training strategy that relies solely on 2D supervision and 3D geometric regularization. Through joint training on object-level and scene-level datasets, our model generalizes across diverse scenes and performs inference in a single forward pass, without any test-time optimization or post-processing. We evaluate our method on diverse tasks including mesh and pointmap reconstruction, novel view synthesis, depth and surface normal estimation. Extensive experiments demonstrate that our model achieves strong performance in consistent and continuous geometry reconstruction from pose-free multi-view inputs.

2 Related Work

Multi-view 3D Reconstruction. Conventional pipelines typically decompose this problem into multiple stages, where sparse geometry and camera poses are first estimated via structure-from-motion (SfM) (Agarwal et al., 2011; Crandall et al., 2012; Wu, 2013; Wilson & Snavely, 2014; Schönberger & Frahm, 2016), followed by dense reconstruction using multi-view stereo (MVS) (Furukawa et al., 2015; Schönberger et al., 2016; Wei et al., 2021; Fu et al., 2022), neural radiance fields (NeRF) (Mildenhall et al., 2021; Barron et al., 2022; Chen et al., 2022), or 3D Gaussian splatting (3DGS) (Kerbl et al., 2023). Recent learning-based methods (Wang et al., 2024a; 2025b; Wu et al., 2025a; Zhuo et al., 2026; Wang et al., 2025a) aim to simplify this process by directly learning geometry from images. They remove the dependency on camera poses by jointly reasoning about multi-view correspondences, enabling end-to-end training and large-scale generalization. However, they primarily focus on predicting geometry in view-aligned explicit forms such as depth maps or point maps, and do not directly model continuous scene representations, limiting their applications to downstream tasks such as novel view synthesis and surface reconstruction.

Neural Implicit Scene Representations. Neural implicit representations model 3D scenes as continuous functions parameterized by neural networks. Early methods (Xie et al., 2022; Chen & Zhang, 2019; Mescheder et al., 2019) rely on an MLP to represent geometry and appearance of a single scene. Subsequent works (Chibane et al., 2020; Martel et al., 2021; Takikawa et al., 2021; Yu et al., 2022) introduce hybrid designs that combine MLPs with multi-resolution voxel grids to capture finer geometric details more efficiently. Most of them are optimized on a per-scene basis and require long training time, which limits their applicability in large-scale or real-time settings. Recent works (Yu et al., 2021; Wang et al., 2021b; Chen et al., 2021; Sun et al., 2021; Guo et al., 2022; Yu et al., 2026) explore generalizable neural fields that can be trained across multiple scenes and adapted to new scenes with minimal finetuning, which enable fast inference and improved generalization across

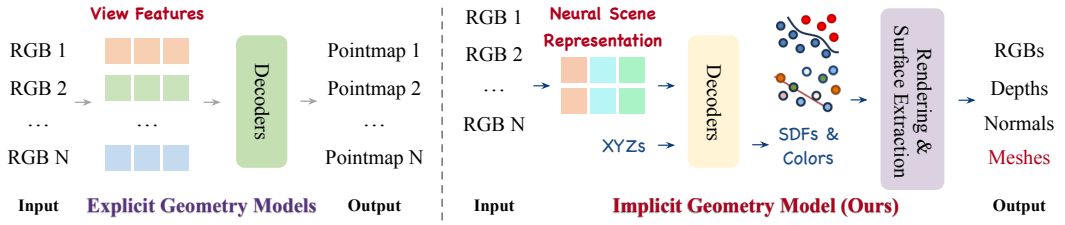

Figure 2: **Explicit vs. implicit visual geometry paradigms.** Existing explicit models decode per-pixel 3D point coordinates for each input view independently, producing discrete and view-indexed pointmaps. Our implicit model instead learns a continuous 3D field from which any spatial location can be queried, enabling direct surface extraction and novel view rendering without post-processing.

scenes. However, they typically rely on known camera poses to establish geometric relationships across views. In addition, since these methods are primarily designed for novel view synthesis, they focus more on appearance modeling than on enforcing consistent multi-view geometry (Jeong et al., 2026), which can limit the quality of reconstructed surfaces.

Surface Reconstruction from Multi-view Images. Existing methods mainly differ in how they represent and recover surfaces. Early methods (Wang et al., 2021a; Guo et al., 2022; Yu et al., 2022; Dong et al., 2023) model a scene as a continuous signed distance field (SDF) whose zero level set defines the surface. While this formulation enables coherent mesh extraction, it is typically optimized for each calibrated scene. Subsequent methods (Sun et al., 2021; Sayed et al., 2022; Stier et al., 2023) fuse pose-aligned multi-view features into a volumetric representation and predict Truncated SDF values for surface reconstruction. Recent generalizable 2D Gaussian splatting (2DGS) methods (Jena et al., 2025; He et al., 2026; Chang et al., 2025; Dai et al., 2026) predict pixel-aligned 2D Gaussian primitives from calibrated multi-view images for feed-forward surface reconstruction. TriSplat (Wang et al., 2026) replaces Gaussian primitives with oriented triangles to support direct mesh concersion. Its representation remains a dense collection of local primitives, which is potentially redundant and does not enforce a globally coherent, watertight surface. Differently, we train an implicit visual geometry transformer that aggregates a neural scene representation from pose-free images, enabling not only continuous spatial queries to render RGB images, depth maps, and surface normal maps from arbitrary viewpoints, but also to extract a continuous surface representation.

3 Proposed Approach

3.1 Implicit Visual Geometry Modeling

Visual geometry foundation models aim to recover the 3D structure of a scene from unposed multi-view images in a single forward pass, without per-scene optimization or known camera parameters. By learning from large-scale multi-view data, such models acquire generalizable geometric priors and enable fast, robust reconstruction across diverse scenes and domains (Wang et al., 2024a; 2025a; Wu et al., 2025a; Zhuo et al., 2026). A central design choice in these models is how scene geometry is represented: the representation determines what geometric queries are supported, whether the output is continuous or discrete, and how rendering and surface extraction can be performed.

Explicit Visual Geometry. As illustrated in Figure 2, existing visual geometry foundation models (Wang et al., 2024a; 2025a; Wu et al., 2025a) follow an **explicit** paradigm, where the model directly decodes a pixel-aligned pointmap for each input view. Formally, given N input images $\{I_i\}_{i=1}^N$, these methods produce a pointmap $\mathcal{P}_i \in \mathbb{R}^{H\times W\times 3}$ per view:

$$\mathcal{P}_i = h_i(\mathcal{F}), \quad \mathcal{F} = \mathrm{Enc}(\{I_i\}_{i=1}^N), \tag{1}$$

where h_i is a per-view decoder and $\mathcal{F}$ is the encoded multi-view feature. The full scene geometry is thus a finite collection of $N \times H \times W$ discrete 3D points, each anchored to a specific pixel in a specific view. This explicit formulation has two fundamental limitations. **First**, the representation is inherently *discrete*: geometry is only defined at pixel-aligned locations, so arbitrary 3D points cannot be queried and no continuous surface is implied. **Second**, the same physical surface point $\mathbf{x}$ may appear in multiple views and be predicted redundantly, yielding potentially inconsistent estimates across views. As a result, extracting a continuous surface or rendering from a novel viewpoint both require additional modeling or non-trivial post-processing.

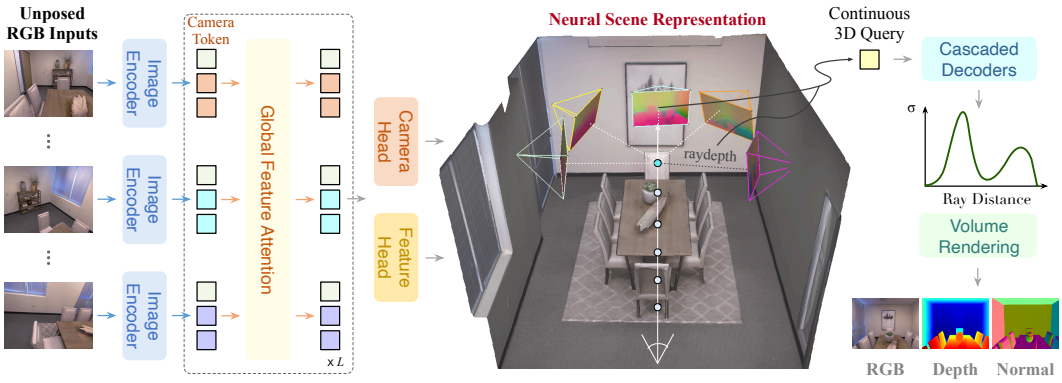

Figure 3: **Framework of IVGT.** Our model takes pose-free multi-view images as input and encodes them into per-view features, which are aggregated via global feature attention into a unified scene representation in a canonical coordinate system. This representation supports continuous 3D queries, where cascaded decoders predict SDF and colors for volume rendering and surface extraction.

Implicit Visual Geometry (Ours). In contrast, we propose to learn a continuous 3D field from pose-free multi-view images. Rather than decoding a fixed set of pixel-aligned points, the model learns a global neural scene representation $\mathcal{F}$ that can be queried at *any* 3D location $\mathbf{x} \in \mathbb{R}^3$:

$$\big(\hat{s}(\mathbf{x}),\, \hat{\mathbf{c}}(\mathbf{x}, \mathbf{v})\big) = \mathrm{Dec}(\mathbf{x},\, \mathbf{v};\, \mathcal{F})\,, \tag{2}$$

where $\hat{s}(\mathbf{x}) \in \mathbb{R}$ is the signed distance value and $\hat{\mathbf{c}}(\mathbf{x}, \mathbf{v}) \in \mathbb{R}^3$ is the view-dependent color for viewing direction $\mathbf{v}$. The scene surface is uniquely defined as the zero level set $\mathcal{S} = \{\mathbf{x} \in \mathbb{R}^3 : \hat{s}(\mathbf{x}) = 0\}$, which is continuous and view-count-independent. Volume rendering along any ray directly yields RGB, depth, and surface normal, and the iso-surface can be extracted via Marching Cubes (Lorensen & Cline, 1998) without post-processing. The full input–output mapping of IVGT is:

$$\big(\mathcal{F},\, (D_i,\, \mathbf{g}_i)_{i=1}^N\big) = f\big(\{I_i\}_{i=1}^N\big)\,, \tag{3}$$

where $\mathbf{g}_i \in \mathbb{R}^9$ and $D_i \in \mathbb{R}^{H \times W}$ are auxiliary camera parameters and depth maps predicted alongside $\mathcal{F}$. We take the first frame as the reference, and the model is permutation equivariant for other views.

3.2 Pose-free Continuous Neural Scene Representation

Global Feature Extraction. As shown in Figure 3, we adopt a transformer-based backbone to extract multi-view image features and learn a global scene representation. Given a set of input images $(I_i)_{i=1}^N$, each image is first converted into a set of tokens by DINO (Oquab et al., 2023). All tokens from different views are then jointly processed by alternating frame-wise and global self-attention (Wang et al., 2025a), enabling both intra-view feature learning and cross-view information aggregation. This design allows the model to implicitly align multi-view observations without explicit camera poses.

The backbone also predicts auxiliary geometric quantities for each view, including depth map D_i and camera parameters $\mathbf{g}_i$, which provide useful geometric cues for downstream representation learning. We obtain a global scene representation $\mathcal{F} = \{F_i\}_{i=1}^N$ from this backbone, where each F_i encodes view-specific features that are implicitly aligned within a shared canonical coordinate system.

Continuous 3D Query and Multi-view Aggregation. For an arbitrary 3D point $\mathbf{x}$ defined in the canonical coordinate system of the first frame, we project it onto all input views using the corresponding predicted camera parameters. Suppose that $\mathbf{x}$ is visible in N_k valid views, we first sample and aggregate the corresponding features from these views:

$$\mathbf{z}_f = \frac{1}{N_k} \sum_{i=1}^{N_k} F_i\big(\pi_i(\mathbf{x})\big). \tag{4}$$

However, points located at different positions along the same projection ray may be mapped to the same pixel and thus share identical image features, while their underlying geometry and appearance can differ significantly. Therefore, it is necessary to augment each queried point with positional information that reflects its spatial location. We observe that what remains invariant is the relative position of a point with respect to each view, which can be represented by its ray depth. We therefore

encode the ray depths of $\mathbf{x}$ with respect to all valid views using a lightweight MLP and aggregate them to obtain a ray-depth feature:

$$\mathbf{z}_d = \frac{1}{N_k} \sum_{i=1}^{N_k} f_{\text{raydepth}}(d_i(\mathbf{x})), \tag{5}$$

where $d_i(\mathbf{x})$ is the ray depth of $\mathbf{x}$ in each valid view and f_{raydepth} denotes this ray-depth embedding. We concatenate the aggregated features $\mathbf{z}_f$ with $\mathbf{z}_d$ to form the local spatial feature $\mathbf{z}$ of $\mathbf{x}$.

Geometry and Appearance Decoding. Given the spatial feature of a queried point, we employ cascaded geometry and appearance decoders to predict its SDF value and color. Specifically, we first use an 8-layer MLP f_θ to decode the SDF value $\hat{s}$ and an intermediate appearance feature $\hat{\mathbf{z}}$:

$$(\hat{s}, \hat{\mathbf{z}}) = f_\theta(\mathbf{z}). \tag{6}$$

Then, we concatenate the appearance feature $\hat{\mathbf{z}}$, the gradient $\hat{\mathbf{n}}$ of the SDF value at this point, and a fixed positional encoding of the corresponding viewing direction $\mathbf{v}$. This concatenated feature is fed into a 2-layer MLP $\mathbf{c}_\theta$ to predict the final color values:

$$\hat{\mathbf{c}} = \mathbf{c}_\theta(\hat{\mathbf{z}}, \hat{\mathbf{n}}, \gamma(\mathbf{v})), \tag{7}$$

where γ denotes the positional encoding following NeRF (Mildenhall et al., 2021).

3.3 Rendering and Surface Extraction

Volume Rendering. We optimize the neural scene representation described in Section 3.2 via a render-based reconstruction loss using differentiable volume rendering (Oechsle et al., 2021; Wang et al., 2021a; Yu et al., 2022). To render a pixel from input or novel viewpoints, we cast a ray $\mathbf{r}$ from the camera center $\mathbf{o}$ through the pixel along its view direction $\mathbf{v}$. Along each ray, we sample M points $\mathbf{x}_\mathbf{r}^i = \mathbf{o} + t_\mathbf{r}^i \mathbf{v}$ and predict their SDF values $\hat{s}_\mathbf{r}^i$ and color values $\hat{\mathbf{c}}_\mathbf{r}^i$. Following VolSDF (Yariv et al., 2021), we transform the SDF values $\hat{s}_\mathbf{r}^i$ to density values $\sigma_\mathbf{r}^i$ for volume rendering:

$$\sigma_\beta(s) = \frac{1}{2\beta} \exp\left(\frac{s}{\beta}\right) \text{ if } s \leq 0, \quad \text{else} \quad \frac{1}{\beta}\left(1 - \frac{1}{2}\exp\left(-\frac{s}{\beta}\right)\right), \tag{8}$$

where β is a learnable parameter shared across different scenes. Following NeRF (Mildenhall et al., 2021), we compute the transmittance $T_\mathbf{r}^i$ and alpha value $\alpha_\mathbf{r}^i$ of sample point i along ray $\mathbf{r}$:

$$T_\mathbf{r}^i = \prod_{j=1}^{i-1} \left(1 - \alpha_\mathbf{r}^j\right) \quad \alpha_\mathbf{r}^i = 1 - \exp\left(-\sigma_\mathbf{r}^i \delta_\mathbf{r}^i\right), \tag{9}$$

where $\delta_\mathbf{r}^i$ is the distance between neighboring sample points. Then we can compute the color $\hat{C}(\mathbf{r})$, the depth $\hat{D}(\mathbf{r})$ and normal $\hat{N}(\mathbf{r})$ of the surface intersecting the current ray as:

$$\hat{C}(\mathbf{r}) = \sum_{i=1}^{M} T_\mathbf{r}^i \alpha_\mathbf{r}^i \hat{\mathbf{c}}_\mathbf{r}^i \quad \hat{D}(\mathbf{r}) = \sum_{i=1}^{M} T_\mathbf{r}^i \alpha_\mathbf{r}^i t_\mathbf{r}^i \quad \hat{N}(\mathbf{r}) = \sum_{i=1}^{M} T_\mathbf{r}^i \alpha_\mathbf{r}^i \hat{\mathbf{n}}_\mathbf{r}^i. \tag{10}$$

Surface Extraction. During inference, we first project the depth maps predicted from the input-view features into the coordinate system of the first frame using the estimated camera poses, obtaining a coarse scene point cloud. The spatial extent of the scene is then computed by the extrema of this point cloud. Next, we uniformly sample a 64^3 grid of points within this bounding volume and infer their SDF values. Points whose absolute SDF values are below a predefined threshold are marked as valid. We then upsample the grid to a higher resolution of 512^3, retain only the valid regions, and evaluate their SDF values for surface reconstruction. The final surface is extracted using Marching Cubes (Lorensen & Cline, 1998). We predict colors for the extracted surface points, and the viewing direction required in Equation (7) is approximated using the normal direction at each surface point.

3.4 Training

We adopt a two-stage training strategy to stabilize the training of our IVGT model. In the first stage, we supervise estimated poses and rendered 2D outputs. We use $\mathcal{L}_{\text{rgb}}$, $\mathcal{L}_{\text{depth}-\text{render}}$, $\mathcal{L}_{\text{normal}}$, and

Table 1: **Scene-level mesh reconstruction on ScanNet.** We compare against per-scene optimization and generalizable baselines using standard mesh reconstruction metrics. Methods marked with * use the poses estimated by our method as input, as they cannot estimate poses from the input images.

	Setting	Acc↓	Comp↓	Chamfer ↓	Prec↑	Recall↑	F-score↑	Time
COLMAP (Schönberger et al., 2016)	Per-Scene	0.047	0.235	0.141	0.711	0.441	0.537	∼1 h
UNISURF (Oechsle et al., 2021)	Per-Scene	0.554	0.164	0.359	0.212	0.362	0.267	∼8 h
NeuS (Wang et al., 2021a)	Per-Scene	0.179	0.208	0.194	0.313	0.275	0.291	∼7 h
VolSDF (Yariv et al., 2021)	Per-Scene	0.414	0.120	0.267	0.321	0.394	0.346	∼8 h
Manhattan-SDF (Guo et al., 2022)	Per-Scene	0.072	0.068	0.070	0.621	0.586	0.602	∼10 h
MonoSDF (Yu et al., 2022)	Per-Scene	**0.035**	**0.048**	**0.042**	**0.799**	**0.681**	**0.733**	∼22 h
SimpleRecon* (Sayed et al., 2022)	Generalizable	0.063	0.097	0.080	0.511	0.495	0.499	∼20 s
FineRecon* (Stier et al., 2023)	Generalizable	0.058	0.053	0.055	0.580	0.615	0.595	∼30 s
MeshSplat* (Chang et al., 2025)	Generalizable	0.092	0.153	0.122	0.306	0.285	0.295	∼57 s
TriSplat (Wang et al., 2026)	Generalizable	0.084	0.121	0.103	0.389	0.543	0.453	∼10 s
IVGT	Generalizable	**0.039**	**0.045**	**0.042**	**0.756**	**0.694**	**0.723**	∼20 s

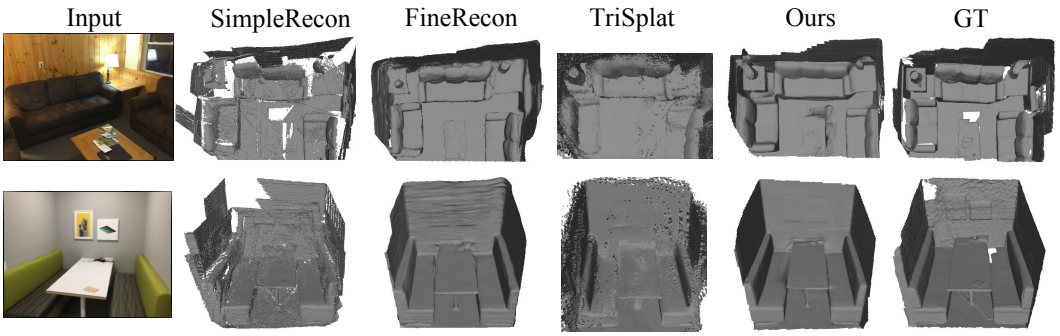

Figure 4: **Qualitative mesh reconstruction on ScanNet.** IVGT produces geometrically complete and surface-coherent meshes in a single forward pass, surpassing other generalizable methods.

$\mathcal{L}_{\text{camera}}$ to denote the RGB reconstruction, rendered depth reconstruction, normal consistency, and camera pose losses. The final loss used in the first stage is

$$\mathcal{L}_{\text{stage1}} = \mathcal{L}_{\text{rgb}} + \lambda_1 \mathcal{L}_{\text{depth-render}} + \lambda_2 \mathcal{L}_{\text{normal}} + \lambda_3 \mathcal{L}_{\text{camera}}. \tag{11}$$

In the second stage, we add an Eiknoal term $\mathcal{L}_{\text{eikonal}}$ (Gropp et al., 2020) and a smoothness term $\mathcal{L}_{\text{smooth}}$ (Oechsle et al., 2021; Yu et al., 2022) to regularize SDF values to improve the stability of geometry reconstruction. To further improve the accuracy of geometry and appearance, we sample some pixels from the ground-truth depth maps and back-project them into the 3D space of the first frame, and then we supervise their SDF values and colors:

$$\mathcal{L}_{\text{point}} = \frac{1}{|\mathcal{P}|} \sum_{\mathbf{x} \in \mathcal{P}} \left(|\hat{s}(\mathbf{x})| + \lambda_{\text{rgb}} \left\| \hat{\mathbf{c}}(\mathbf{x}) - \mathbf{c} \right\|_1 \right), \tag{12}$$

where $\mathcal{P}$ denotes the set of sampled surface points. And we also use the depth loss from VGGT $\mathcal{L}_{\text{depth}}$ (Wang et al., 2025a) to supervise the depth maps directly decoded from the input image features. The final loss in the second stage is:

$$\mathcal{L}_{\text{stage2}} = \mathcal{L}_{\text{stage1}} + \lambda_4 \mathcal{L}_{\text{eikonal}} + \lambda_5 \mathcal{L}_{\text{smooth}} + \lambda_6 \mathcal{L}_{\text{point}} + \lambda_7 \mathcal{L}_{\text{depth}}. \tag{13}$$

We provide the detailed formulations of all loss terms in Section A.

4 Experiments

4.1 Experimental Settings

Implementation Details. We initialize the image encoder and global feature extraction module with pre-trained weights from VGGT (Wang et al., 2025a) and train our model on 4 A800 NVIDIA GPUs for 7 days. We provide more details in Section B.1.

Datasets. We train our IVGT using a diverse collection of 11 datasets covering objects and scenes and including real and synthetic data. We provide a complete list of all datasets in Section B.2.

Table 2: **Pointmap reconstruction on 7-Scenes, NRGBD, and DTU.** We report mean and median Accuracy and Completeness (in meters for scene-level, mm for object-level). Pointmaps decoded directly from per-view features (**IVGT**) consistently outperform prior methods, while pointmaps projected from rendered depth maps (**IVGT** (from render)) also achieve competitive results.

Method	**7-Scenes** (scene, kf=50)				**NRGBD** (scene, kf=100)				**DTU** (object, kf=5)			
	Acc. ↓		Comp. ↓		Acc. ↓		Comp. ↓		Acc. ↓		Comp. ↓	
	Mean	Med.	Mean	Med.	Mean	Med.	Mean	Med.	Mean	Med.	Mean	Med.
Fast3R (Yang et al., 2025)	0.053	0.023	0.084	0.033	0.080	0.039	0.075	0.039	3.605	1.855	3.048	1.274
CUT3R (Wang et al., 2025b)	0.024	0.011	0.028	0.009	0.086	0.038	0.047	0.016	4.749	2.588	2.764	1.176
Point3R (Wu et al., 2025a)	0.030	0.015	0.026	0.010	0.065	0.035	0.030	0.013	6.111	3.930	2.880	1.399
StreamVGGT (Zhuo et al., 2026)	0.046	0.025	0.032	0.014	0.080	0.056	0.044	0.021	2.520	1.395	2.048	1.075
VGGT (Wang et al., 2025a)	0.020	0.008	0.028	0.012	0.018	**0.010**	0.017	0.009	**1.305**	**0.743**	2.435	1.737
IVGT	**0.015**	**0.007**	**0.019**	**0.008**	**0.017**	0.011	**0.015**	**0.006**	1.885	1.078	**1.595**	**0.844**
IVGT (from render)	0.019	0.008	0.030	0.016	0.038	0.020	0.025	0.013	4.715	2.753	2.620	1.048



Figure 5: **Mesh vs. pointmap representations.** Pixel-aligned pointmap reconstruction suffers from sparsity and surface discontinuities, especially at object boundaries. Meshes extracted from IVGT exhibit significantly improved geometric continuity and completeness, validating the advantage of continuous implicit geometry over discrete explicit representations.

4.2 3D Geometry Reconstruction

3D Mesh Reconstruction. We first compare against various baselines to show the performance of IVGT for mesh reconstruction. We use the test split from **ScanNet** (Dai et al., 2017) following Manhattan-SDF (Guo et al., 2022) and MonoSDF (Yu et al., 2022). In Table 1, we report the average reconstruction results and optimization or inference time for IVGT and baselines on several test scenes. Our method outperforms other generalizable methods and most per-scene optimization methods, achieving performance comparable to MonoSDF. Given that MonoSDF requires several hours of optimization for each scene, IVGT offers a superior trade-off between computational efficiency and reconstruction quality. We provide more details in Section B.3. Qualitative comparisons with several generalizable baselines are shown in Figure 4.

3D Pointmap Reconstruction. We evaluate the quality of pointmap reconstruction on both scene-level and object-level datasets, including **7-Scenes** (Shotton et al., 2013), **NRGBD** (Azinović et al., 2022), and **DTU** (Jensen et al., 2014) in Table 2. IVGT denotes that we reconstruct pointmaps from the directly decoded depths and poses, and IVGT (fomr render) means that we render depth maps for the input viewpoints. Note that we only provide comparisons with a few representative methods for reference, as pointmap reconstruction is not the primary focus of IVGT but rather a by-product of our representation. We observe that the point maps reconstructed from depth maps achieve better performance than previous methods. The point maps obtained by projecting rendered depth maps also achieve results comparable to these representative methods.

Qualitative Comparison. We further compare mesh and pointmap representations in Figure 5. Pointmap reconstruction often suffers from sparsity and lack of surface continuity. In contrast, meshes extracted from our implicit representation exhibit significantly improved surface coherence and geometric completeness. This demonstrates that IVGT enables not only accurate point-level reconstruction but also high-quality continuous surface modeling.

Camera Pose Estimation. Following MonST3R (Zhang et al., 2024) and CUT3R (Wang et al., 2025b), we evaluate the camera pose estimation performance on **ScanNet**, **Sintel** and **TUM-dynamics**, as shown in Table 3. We report Absolute Translation Error (ATE), Relative Translation Error (RPE trans), and Relative Rotation Error (RPE rot) after applying a Sim(3) Umeyama alignment with the ground truth. Our method achieves comparable or even superior performance to representative feed-forward approaches on camera pose estimation metrics across multiple datasets.

Table 3: **Camera pose estimation on ScanNet, Sintel, and TUM-dynamics.** We report Absolute Translation Error (ATE), Relative Translation Error (RPE trans), and Relative Rotation Error (RPE rot) after Sim(3) Umeyama alignment. IVGT achieves competitive or superior performance to state-of-the-art feed-forward methods across all three benchmarks.

Method	ScanNet (Static)			Sintel			TUM-dynamics		
	ATE↓	RPE trans↓	RPE rot↓	ATE↓	RPE trans↓	RPE rot↓	ATE↓	RPE trans↓	RPE rot↓
Fast3R (Yang et al., 2025)	0.084	0.059	2.844	0.274	0.224	15.69	0.048	0.052	1.197
CUT3R (Wang et al., 2025b)	0.096	0.022	0.733	0.210	0.071	0.627	0.045	0.014	0.441
FLARE (Zhang et al., 2025)	0.069	0.025	0.991	0.158	0.082	2.863	0.026	0.013	0.475
VGGT (Wang et al., 2025a)	0.035	0.015	**0.403**	0.169	0.064	**0.478**	**0.012**	**0.010**	0.313
WorldMirror (Liu et al., 2025)	0.037	0.017	0.422	**0.121**	**0.063**	0.535	**0.012**	**0.010**	**0.301**
IVGT	**0.030**	**0.013**	**0.403**	0.135	0.064	0.492	0.015	**0.010**	0.325

Table 4: **Novel view synthesis on RealEstate10K and DL3DV.** We evaluate under sparse-view (2/8 views) and dense-view (32/64 views) settings and report PSNR, SSIM, and LPIPS. IVGT achieves performance comparable to existing feed-forward methods.

Method	RealEstate10K (2 views)			DL3DV (8 views)			RealEstate10K (32 views)			DL3DV (64 views)		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
FLARE (Zhang et al., 2025)	16.33	0.574	0.410	15.35	0.516	0.591	–	–	–	–	–	–
AnySplat (Jiang et al., 2025)	17.62	0.616	**0.242**	18.31	0.569	**0.258**	19.96	0.718	0.234	18.40	0.602	**0.286**
TriSplat (Wang et al., 2026)	18.27	0.641	0.277	17.03	0.465	0.362	**24.01**	**0.831**	**0.107**	17.11	0.489	0.394
IVGT	**18.85**	**0.652**	0.440	**19.64**	**0.625**	0.489	21.79	0.745	0.390	**19.63**	**0.654**	0.499

Table 5: **Depth estimation on NYUv2 and Sintel.** IVGT achieves competitive or superior depth metrics compared with the baselines.

Method	NYUv2		Sintel	
	Abs Rel↓	$\delta < 1.25$ ↑	Abs Rel↓	$\delta < 1.25$ ↑
Fast3R (Yang et al., 2025)	0.093	0.898	0.544	0.509
CUT3R (Wang et al., 2025b)	0.081	0.914	0.418	0.520
VGGT (Wang et al., 2025a)	**0.056**	0.951	0.606	0.599
IVGT	0.061	**0.955**	**0.293**	**0.636**
IVGT (from render)	0.065	0.950	0.333	0.616

Table 6: **Surface normal estimation on NYUv2 and iBims-1.** IVGT achieves results comparable to dedicated normal estimation methods.

Method	NYUv2			iBims-1		
	mean↓	med↓	22.5° ↑	mean↓	med↓	22.5° ↑
OASIS (Chen et al., 2020)	29.2	23.4	48.4	32.6	24.6	46.6
EESNU (Bae et al., 2021)	**16.2**	8.5	77.2	20.0	8.4	73.4
Omnidata v2 (Kar et al., 2022)	17.2	9.7	76.5	18.2	7.0	77.4
DSine (Bae & Davison, 2024)	16.4	**8.4**	77.7	**17.1**	**6.1**	**79.0**
IVGT	**16.2**	9.9	**78.1**	19.5	10.2	75.3

4.3 Novel View Rendering and Geometry Estimation

Novel View Synthesis. We use Table 4 to report the quantitative evaluation results for novel view synthesis under the no-pose feed-forward setting. For **RealEstate10K** (Zhou et al., 2018), we use 200 sampled test scenes following NopoSplat (Ye et al., 2024). We use 3 novel views per scene in the sparse-view setting and 4 novel views per scene in the dense-view setting. For **DL3DV** (Ling et al., 2024), we follow the FLARE (Zhang et al., 2025) test split and evaluate in 112 unseen scenes. Each test scene contains 9 novel views in both sparse-view and dense-view settings. Our method achieves performance comparable to existing feed-forward methods on this task.

Depth Estimation. We compare with several representative feed-forward methods including in Table 5. We evaluate the depth estimation on **NYUv2** (Silberman et al., 2012) and **Sintel** (Butler et al., 2012) datasets. We report the absolute relative error (Abs Rel) and threshold accuracy ($\delta < 1.25$). The depth maps directly decoded from multi-view features (IVGT) achieve better performance in depth estimation metrics than prior methods, while the depth maps rendered from the neural scene representation (IVGT (from render)) obtain comparable results under the same metrics, demonstrating that the implicit representation preserves accurate geometry.

Surface Normal Estimation. Following DSine (Bae & Davison, 2024), we evaluate surface normal estimation on **NYUv2** and **iBims-1**, as shown in Table 6. We report the angular error between predicted and ground-truth normal maps, including mean and median errors, as well as the percentage of pixels with angular error below 22.5°. Our results are comparable to these strong baselines, demonstrating that our model can recover consistent surface geometry despite being primarily optimized for multi-view 3D reconstruction rather than dedicated normal estimation.

4.4 Experimental Analysis

Visualizations. We visualize our mesh reconstruction results on diverse scenes and objects, as shown in Figure 6. We also visualize the novel view rendering results on the **ScanNet** test scenes in Figure 7.



Figure 6: **Colored mesh reconstruction on diverse scenes and objects.** IVGT produces complete and consistent colored meshes on scenes and objects.

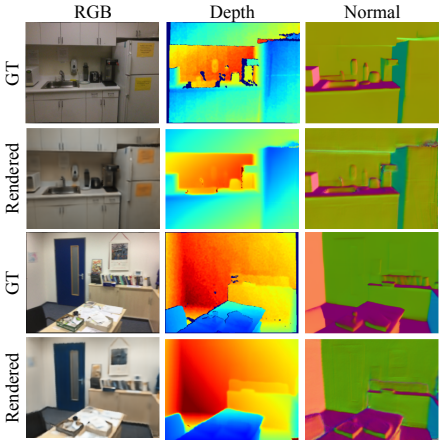

Figure 7: **IVGT renders RGBs, depths, and surface normals using the learned scene representation.**

Table 7: **Ablation on positional encoding strategies for 3D query points.**

	Acc↓	Comp↓	Chamfer↓	Prec↑	Recall↑	F-score↑
XYZ-PosEnc	0.624	0.158	0.391	0.220	0.346	0.260
XYZ-Embed	0.494	0.148	0.321	0.272	0.364	0.306
Raydepth-Embed	**0.219**	**0.103**	**0.161**	**0.434**	**0.431**	**0.429**

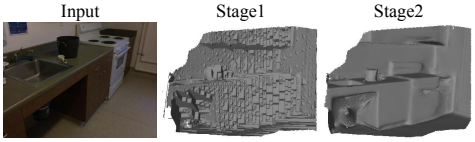

Figure 8: **Effect of the training strategy.**

Ablation on the Positional Information Injecting Choices. Directly encoding positional information using absolute 3D coordinates may introduce ambiguity due to the dependence on the choice of reference frame. To further validate the effectiveness of our ray-depth embedding in Section 3.2, we conduct an ablation study on ScanNet, where we train four variants with different positional information injecting strategies under the same training setup. Each setting is trained for 16 hours. The variants include *No Embed*, fixed coordinate positional encoding (*XYZ-PosEnc*), MLP-based coordinate embedding (*XYZ-Embed*), and our *Raydepth-Embed*. We evalute the mesh reconstruction performance of these four variants on several test scenes we use in Section 4.2 and report the average metrics in Table 7. *No Embed* fails to reconstruct meaningful geometry, producing mostly positive SDF values and no valid mesh. Both coordinate-based variants yield limited performance, likely due to ambiguity introduced by reference-frame-dependent representations. In contrast, *Raydepth-Embed* achieves the best results on all metrics by encoding view-consistent relative geometry.

Analysis on the Training Setup. We adopt a two-stage training strategy to balance geometric learning and surface regularization. Directly introducing 3D constraints in the early stage tends to hinder the model from learning the underlying geometric distribution, while relying solely on 2D supervision leads to rough and irregular surfaces. We therefore first train the model using only 2D supervision to capture coarse geometry, and then introduce Eikonal and smoothness regularization in the second stage to enforce surface consistency. As shown in Figure 8, the first stage results exhibit noticeable surface irregularities, whereas the second stage produces significantly smoother and more coherent surfaces, validating the effectiveness of our training setup.

5 Conclusion

In this paper, we have presented IVGT, an implicit visual geometry transformer that learns a pose-free neural scene representation and enables continuous spatial queries for joint geometry and appearance modeling. Unlike prior explicit models that decode discrete, pixel-aligned pointmaps, IVGT models the scene as a global implicit SDF field that supports continuous 3D queries at arbitrary locations. Extensive experiments demonstrate that our method achieves strong performance across multiple tasks, including mesh reconstruction, point cloud reconstruction, novel view synthesis, depth and surface normal estimation, and camera pose estimation, while generalizing across diverse scenes. Our formulation bridges feed-forward multi-view representation learning with neural implicit geometry modeling, representing a step toward scalable and pose-free continuous surface reconstruction.

AI USE STATEMENT

In this work, we have not used generative AI tools for any tasks with required disclosure, and all such tasks are not applicable to this work. We used generative AI tools only to polish the writing of parts of the manuscript. The authors reviewed all AI-assisted edits to ensure that they preserve the intended technical meaning and contain no unsupported claims. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

To facilitate reproducibility, the main paper, Section A and Section B provide detailed descriptions of our methodology and implementation. We will release the complete source code and model weights.

REFERENCES

Sameer Agarwal, Yasutaka Furukawa, Noah Snavely, Ian Simon, Brian Curless, Steven M Seitz, and Richard Szeliski. Building rome in a day. *Communications of the ACM*, 54(10):105–112, 2011.

Dejan Azinović, Ricardo Martin-Brualla, Dan B Goldman, Matthias Nießner, and Justus Thies. Neural rgb-d surface reconstruction. In *CVPR*, pp. 6290–6301, 2022.

Gwangbin Bae and Andrew J Davison. Rethinking inductive biases for surface normal estimation. In *CVPR*, pp. 9535–9545, 2024.

Gwangbin Bae, Ignas Budvytis, and Roberto Cipolla. Estimating and exploiting the aleatoric uncertainty in surface normal estimation. In *ICCV*, pp. 13137–13146, 2021.

Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *CVPR*, pp. 5470–5479, 2022.

Gilad Baruch, Zhuoyuan Chen, Afshin Dehghan, Tal Dimry, Yuri Feigin, Peter Fu, Thomas Gebauer, Brandon Joffe, Daniel Kurz, Arik Schwartz, et al. Arkitscenes: A diverse real-world dataset for 3d indoor scene understanding using mobile rgb-d data. *arXiv preprint arXiv:2111.08897*, 2021.

Daniel J Butler, Jonas Wulff, Garrett B Stanley, and Michael J Black. A naturalistic open source movie for optical flow evaluation. In *ECCV*, pp. 611–625. Springer, 2012.

Hanzhi Chang, Ruijie Zhu, Wenjie Chang, Mulin Yu, Yanzhe Liang, Jiahao Lu, Zhuoyuan Li, and Tianzhu Zhang. Meshsplat: Generalizable sparse-view surface reconstruction via gaussian splatting. *arXiv preprint arXiv:2508.17811*, 2025.

Anpei Chen, Zexiang Xu, Fuqiang Zhao, Xiaoshuai Zhang, Fanbo Xiang, Jingyi Yu, and Hao Su. Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo. In *ICCV*, pp. 14124–14133, 2021.

Anpei Chen, Zexiang Xu, Andreas Geiger, Jingyi Yu, and Hao Su. Tensorf: Tensorial radiance fields. In *ECCV*, pp. 333–350. Springer, 2022.

Weifeng Chen, Shengyi Qian, David Fan, Noriyuki Kojima, Max Hamilton, and Jia Deng. Oasis: A large-scale dataset for single image 3d in the wild. In *CVPR*, pp. 679–688, 2020.

Xingyu Chen, Yue Chen, Yuliang Xiu, Andreas Geiger, and Anpei Chen. Ttt3r: 3d reconstruction as test-time training. *arXiv preprint arXiv:2509.26645*, 2025.

Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *CVPR*, pp. 5939–5948, 2019.

Julian Chibane, Thiemo Alldieck, and Gerard Pons-Moll. Implicit functions in feature space for 3d shape reconstruction and completion. In *CVPR*, pp. 6970–6981, 2020.

David J Crandall, Andrew Owens, Noah Snavely, and Daniel P Huttenlocher. Sfm with mrfs: Discrete-continuous optimization for large-scale structure from motion. *TPAMI*, 35(12):2841–2853, 2012.

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

Chensheng Dai, Shengjun Zhang, Min Chen, and Yueqi Duan. Surfelsplat: Learning efficient and generalizable gaussian surfel representations for sparse-view surface reconstruction. *NeurIPS*, 38: 173477–173504, 2026.

Wei Dong, Christopher Choy, Charles Loop, Or Litany, Yuke Zhu, and Anima Anandkumar. Fast monocular scene reconstruction with global-sparse local-dense grids. In *CVPR*, pp. 4263–4272, 2023.

Xianze Fang, Jingnan Gao, Zhe Wang, Zhuo Chen, Xingyu Ren, Jiangjing Lyu, Qiaomu Ren, Zhonglei Yang, Xiaokang Yang, Yichao Yan, et al. Dens3r: A foundation model for 3d geometry prediction. *arXiv preprint arXiv:2507.16290*, 2025.

Qiancheng Fu, Qingshan Xu, Yew Soon Ong, and Wenbing Tao. Geo-neus: Geometry-consistent neural implicit surfaces learning for multi-view reconstruction. *NeurIPS*, 35:3403–3416, 2022.

Yasutaka Furukawa and Jean Ponce. Accurate, dense, and robust multiview stereopsis. *TPAMI*, 32 (8):1362–1376, 2009.

Yasutaka Furukawa, Carlos Hernández, et al. Multi-view stereo: A tutorial. *Foundations and Trends® in Computer Graphics and Vision*, 9(1-2):1–148, 2015.

Amos Gropp, Lior Yariv, Niv Haim, Matan Atzmon, and Yaron Lipman. Implicit geometric regularization for learning shapes. *arXiv preprint arXiv:2002.10099*, 2020.

Haoyu Guo, Sida Peng, Haotong Lin, Qianqian Wang, Guofeng Zhang, Hujun Bao, and Xiaowei Zhou. Neural 3d scene reconstruction with the manhattan-world assumption. In *CVPR*, pp. 5511–5520, 2022.

Bing He, Jingnan Gao, Yunuo Chen, Ning Cao, Gang Chen, Zhengxue Cheng, Li Song, and Wenjun Zhang. Surfsplat: Conquering feedforward 2d gaussian splatting with surface continuity priors. In *ICLR*, 2026.

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

Yuanhui Huang, Wenzhao Zheng, Yunpeng Zhang, Jie Zhou, and Jiwen Lu. Tri-perspective view for vision-based 3d semantic occupancy prediction. In *CVPR*, pp. 9223–9232, 2023.

Shubhendu Jena, Shishir Reddy Vutukur, and Adnane Boukhayma. Sparsplat: Fast multi-view reconstruction with generalizable 2d gaussian splatting. *arXiv preprint arXiv:2505.02175*, 2025.

Rasmus Jensen, Anders Dahl, George Vogiatzis, Engil Tola, and Henrik Aanæs. Large scale multi-view stereopsis evaluation. In *CVPR*, pp. 406–413. IEEE, 2014.

Hyeonjun Jeong, Juyeb Shin, and Dongsuk Kum. To view transform or not to view transform: Nerf-based pre-training perspective. *arXiv preprint arXiv:2603.28090*, 2026.

Lihan Jiang, Yucheng Mao, Linning Xu, Tao Lu, Kerui Ren, Yichen Jin, Xudong Xu, Mulin Yu, Jiangmiao Pang, Feng Zhao, et al. Anysplat: Feed-forward 3d gaussian splatting from unconstrained views. *TOG*, 44(6):1–16, 2025.

Oğuzhan Fatih Kar, Teresa Yeo, Andrei Atanov, and Amir Zamir. 3d common corruptions and data augmentation. In *CVPR*, pp. 18963–18974, 2022.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *TOG*, 42(4), July 2023.

Yushi Lan, Yihang Luo, Fangzhou Hong, Shangchen Zhou, Honghua Chen, Zhaoyang Lyu, Shuai Yang, Bo Dai, Chen Change Loy, and Xingang Pan. Stream3r: Scalable sequential 3d reconstruction with causal transformer. *arXiv preprint arXiv:2508.10893*, 2025.

Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *CVPR*, pp. 2041–2050, 2018.

Lu Ling, Yichen Sheng, Zhi Tu, Wentian Zhao, Cheng Xin, Kun Wan, Lantao Yu, Qianyu Guo, Zixun Yu, Yawen Lu, et al. Dl3dv-10k: A large-scale scene dataset for deep learning-based 3d vision. In *CVPR*, pp. 22160–22169, 2024.

Yifan Liu, Zhiyuan Min, Zhenwei Wang, Junta Wu, Tengfei Wang, Yixuan Yuan, Yawei Luo, and Chunchao Guo. Worldmirror: Universal 3d world reconstruction with any-prior prompting. *arXiv preprint arXiv:2510.10726*, 2025.

William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *Seminal graphics: pioneering efforts that shaped the field*, pp. 347–353. 1998.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Julien NP Martel, David B Lindell, Connor Z Lin, Eric R Chan, Marco Monteiro, and Gordon Wetzstein. Acorn: Adaptive coordinate networks for neural scene representation. *arXiv preprint arXiv:2105.02788*, 2021.

Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *CVPR*, pp. 4460–4470, 2019.

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

Michael Oechsle, Songyou Peng, and Andreas Geiger. Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In *ICCV*, pp. 5589–5599, 2021.

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.

Jeremy Reizenstein, Roman Shapovalov, Philipp Henzler, Luca Sbordone, Patrick Labatut, and David Novotny. Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In *ICCV*, pp. 10901–10911, 2021.

Mike Roberts, Jason Ramapuram, Anurag Ranjan, Atulit Kumar, Miguel Angel Bautista, Nathan Paczan, Russ Webb, and Joshua M Susskind. Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding. In *ICCV*, pp. 10912–10922, 2021.

Mohamed Sayed, John Gibson, Jamie Watson, Victor Prisacariu, Michael Firman, and Clément Godard. Simplerecon: 3d reconstruction without 3d convolutions. In *ECCV*, pp. 1–19, 2022.

Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *CVPR*, 2016.

Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *ECCV*, 2016.

Jamie Shotton, Ben Glocker, Christopher Zach, Shahram Izadi, Antonio Criminisi, and Andrew Fitzgibbon. Scene coordinate regression forests for camera relocalization in rgb-d images. In *CVPR*, pp. 2930–2937, 2013.

Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, pp. 746–760. Springer, 2012.

Noah Snavely, Steven M Seitz, and Richard Szeliski. Photo tourism: exploring photo collections in 3d. In *ACM siggraph 2006 papers*, pp. 835–846. 2006.

Noah Stier, Anurag Ranjan, Alex Colburn, Yajie Yan, Liang Yang, Fangchang Ma, and Baptiste Angles. Finerecon: Depth-aware feed-forward network for detailed 3d reconstruction. In *ICCV*, pp. 18377–18386. IEEE, 2023.

Jiaming Sun, Yiming Xie, Linghao Chen, Xiaowei Zhou, and Hujun Bao. Neuralrecon: Real-time coherent 3d reconstruction from monocular video. In *CVPR*, pp. 15598–15607, 2021.

Towaki Takikawa, Joey Litalien, Kangxue Yin, Karsten Kreis, Charles Loop, Derek Nowrouzezahrai, Alec Jacobson, Morgan McGuire, and Sanja Fidler. Neural geometric level of detail: Real-time rendering with implicit 3d shapes. In *CVPR*, pp. 11358–11367, 2021.

Fabio Tosi, Yiyi Liao, Carolin Schmitt, and Andreas Geiger. Smd-nets: Stereo mixture density networks. In *CVPR*, pp. 8942–8952, 2021.

Hengyi Wang and Lourdes Agapito. 3d reconstruction with spatial memory. *arXiv preprint arXiv:2408.16061*, 2024.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *CVPR*, pp. 5294–5306, 2025a.

Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *arXiv preprint arXiv:2106.10689*, 2021a.

Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul P Srinivasan, Howard Zhou, Jonathan T Barron, Ricardo Martin-Brualla, Noah Snavely, and Thomas Funkhouser. Ibrnet: Learning multi-view image-based rendering. In *CVPR*, pp. 4690–4699, 2021b.

Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state. In *CVPR*, pp. 10510–10522. IEEE, 2025b.

Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *CVPR*, pp. 20697–20709, 2024a.

Tai Wang, Xiaohan Mao, Chenming Zhu, Runsen Xu, Ruiyuan Lyu, Peisen Li, Xiao Chen, Wenwei Zhang, Kai Chen, Tianfan Xue, et al. Embodiedscan: A holistic multi-modal 3d perception suite towards embodied ai. In *CVPR*, pp. 19757–19767, 2024b.

Weijie Wang, Zimu Li, Jinchuan Shi, Zeyu Zhang, Botao Ye, Marc Pollefeys, Donny Y Chen, and Bohan Zhuang. Trisplat: Simulation-ready feed-forward 3d scene reconstruction. *arXiv preprint arXiv:2605.26115*, 2026.

Wenshan Wang, Delong Zhu, Xiangwei Wang, Yaoyu Hu, Yuheng Qiu, Chen Wang, Yafei Hu, Ashish Kapoor, and Sebastian Scherer. Tartanair: A dataset to push the limits of visual slam. In *IROS*, pp. 4909–4916. IEEE, 2020.

Yi Wei, Shaohui Liu, Yongming Rao, Wang Zhao, Jiwen Lu, and Jie Zhou. Nerfingmvs: Guided optimization of neural radiance fields for indoor multi-view stereo. In *ICCV*, pp. 5610–5619, October 2021.

Kyle Wilson and Noah Snavely. Robust global translations with 1dsfm. In *ECCV*, pp. 61–75. Springer, 2014.

Changchang Wu. Towards linear-time incremental structure from motion. In *2013 International Conference on 3D Vision-3DV 2013*, pp. 127–134. IEEE, 2013.

Tong Wu, Jiarui Zhang, Xiao Fu, Yuxin Wang, Jiawei Ren, Liang Pan, Wayne Wu, Lei Yang, Jiaqi Wang, Chen Qian, et al. Omniobject3d: Large-vocabulary 3d object dataset for realistic perception, reconstruction and generation. In *CVPR*, pp. 803–814, 2023.

Yuqi Wu, Wenzhao Zheng, Jie Zhou, and Jiwen Lu. Point3r: Streaming 3d reconstruction with explicit spatial pointer memory. In *NeurIPS*, 2025a.

Yuqi Wu, Wenzhao Zheng, Sicheng Zuo, Yuanhui Huang, Jie Zhou, and Jiwen Lu. Embodiedocc: Embodied 3d occupancy prediction for vision-based online scene understanding. In *ICCV*, 2025b.

Hongchi Xia, Yang Fu, Sifei Liu, and Xiaolong Wang. Rgbd objects in the wild: scaling real-world 3d object learning from rgb-d videos. In *CVPR*, pp. 22378–22389, 2024.

Yiheng Xie, Towaki Takikawa, Shunsuke Saito, Or Litany, Shiqin Yan, Numair Khan, Federico Tombari, James Tompkin, Vincent Sitzmann, and Srinath Sridhar. Neural fields in visual computing and beyond. In *Computer graphics forum*, pp. 641–676. Wiley Online Library, 2022.

Jianing Yang, Alexander Sax, Kevin J Liang, Mikael Henaff, Hao Tang, Ang Cao, Joyce Chai, Franziska Meier, and Matt Feiszli. Fast3r: Towards 3d reconstruction of 1000+ images in one forward pass. *arXiv preprint arXiv:2501.13928*, 2025.

Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. *NeurIPS*, 34:4805–4815, 2021.

Botao Ye, Sifei Liu, Haofei Xu, Xueting Li, Marc Pollefeys, Ming-Hsuan Yang, and Songyou Peng. No pose, no problem: Surprisingly simple 3d gaussian splats from sparse unposed images. *arXiv preprint arXiv:2410.24207*, 2024.

Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *ICCV*, pp. 12–22, 2023.

Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *CVPR*, pp. 4578–4587, 2021.

Hao Yu, Haotong Lin, Jiawei Wang, Jiaxin Li, Yida Wang, Xueyang Zhang, Yue Wang, Xiaowei Zhou, Ruizhen Hu, and Sida Peng. Infinidepth: Arbitrary-resolution and fine-grained depth estimation with neural implicit fields. *arXiv preprint arXiv:2601.03252*, 2026.

Zehao Yu, Songyou Peng, Michael Niemeyer, Torsten Sattler, and Andreas Geiger. Monosdf: Exploring monocular geometric cues for neural implicit surface reconstruction. *NeurIPS*, 35: 25018–25032, 2022.

Junyi Zhang, Charles Herrmann, Junhwa Hur, Varun Jampani, Trevor Darrell, Forrester Cole, Deqing Sun, and Ming-Hsuan Yang. Monst3r: A simple approach for estimating geometry in the presence of motion. *arXiv preprint arXiv:2410.03825*, 2024.

Shangzhan Zhang, Jianyuan Wang, Yinghao Xu, Nan Xue, Christian Rupprecht, Xiaowei Zhou, Yujun Shen, and Gordon Wetzstein. Flare: Feed-forward geometry, appearance and camera estimation from uncalibrated sparse views. In *CVPR*, pp. 21936–21947, 2025.

Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: Learning view synthesis using multiplane images. *arXiv preprint arXiv:1805.09817*, 2018.

Dong Zhuo, Wenzhao Zheng, Jiahe Guo, Yuqi Wu, Jie Zhou, and Jiwen Lu. Streaming 4d visual geometry transformer. *ICLR*, 2026.

Sicheng Zuo, Zixun Xie, Wenzhao Zheng, Shaoqing Xu, Fang Li, Shengyin Jiang, Long Chen, Zhi-Xin Yang, and Jiwen Lu. Dvgt: Driving visual geometry transformer. In *CVPR*, 2026.

Appendix

A Additional Method Details

The loss terms used in the first stage are defined as

$$\begin{aligned}
\mathcal{L}_{\text{rgb}} &= \frac{1}{|\mathcal{R}|}\sum_{\mathbf{r}\in\mathcal{R}}\left\|\hat{C}(\mathbf{r})-C(\mathbf{r})\right\|_1, \\
\mathcal{L}_{\text{depth-render}} &= \frac{1}{|\mathcal{R}|}\sum_{\mathbf{r}\in\mathcal{R}}\left\|\hat{D}(\mathbf{r})-D(\mathbf{r})\right\|_1, \\
\mathcal{L}_{\text{normal}} &= \frac{1}{|\mathcal{R}|}\sum_{\mathbf{r}\in\mathcal{R}}\left(\left\|\hat{N}(\mathbf{r})-\bar{N}(\mathbf{r})\right\|_1+\left\|1-\hat{N}(\mathbf{r})^\top\bar{N}(\mathbf{r})\right\|_1\right), \\
\mathcal{L}_{\text{camera}} &= \frac{1}{N}\sum_{i=1}^{N}|\hat{\mathbf{g}}_i-\mathbf{g}_i|_\epsilon .
\end{aligned} \tag{14}$$

Here, $\mathcal{R}$ denotes the set of pixels in the sampled batch, $C(\mathbf{r})$ is the observed pixel color, $D(\mathbf{r})$ is the ground-truth depth normalized over the entire input sequence, and $\bar{N}(\mathbf{r})$ denotes the predicted monocular normals transformed to the first-frame coordinate system. $\mathbf{g}_i$ is the ground-truth camera parameters, and $|\cdot|_\epsilon$ denotes the Huber loss.

The Eiknoal term and the smoothness term used in the second stage are defined as

$$\begin{aligned}
\mathcal{L}_{\text{eikonal}} &= \frac{1}{|\mathcal{X}|}\sum_{\mathbf{x}\in\mathcal{X}}\left(\|\nabla f_\theta(\mathbf{x})\|_2-1\right)^2, \\
\mathcal{L}_{\text{smooth}} &= \frac{1}{|\mathcal{X}|}\sum_{\mathbf{x}\in\mathcal{X}}\left\|\frac{\nabla f_\theta(\mathbf{x})}{\|\nabla f_\theta(\mathbf{x})\|_2}-\frac{\nabla f_\theta(\mathbf{x}^+)}{\|\nabla f_\theta(\mathbf{x}^+)\|_2}\right\|_2 .
\end{aligned} \tag{15}$$

where $\mathcal{X}$ are a set of uniformly sampled points together with near-surface points (Yariv et al., 2021) and $\mathbf{x}^+$ are neighboring points after adding small perturbations to the original sampled points.

The depth loss (Wang et al., 2025a) we use to supervise the depth maps directly decoded from the input image features are defined as

$$\mathcal{L}_{\text{depth}} = \sum_{i=1}^{N}\|\Sigma_i^D\odot\left(\hat{D}_i-D_i\right)\|+\|\Sigma_i^D\odot\left(\nabla\hat{D}_i-\nabla D_i\right)\|-\alpha\log\Sigma_i^D, \tag{16}$$

where Σ_i^D is the predicted uncertainty map between the predicted depth $\hat{D}_i$ and the ground-truth depth D_i, and $\odot$ is the channel-broadcast element-wise product.

B Additional Experimental Details

B.1 Implementation Details

Architecture. All input images are resized and cropped to a fixed resolution of 518×392. We initialize the image encoder and global feature extraction module with pre-trained weights from VGGT (Wang et al., 2025a). To obtain dense global features for continuous querying, we attach a DPT-style upsampling head to the aggregated tokens from our transformer-based backbone. The head outputs a 128-dimensional feature map at the input image resolution. We initialize this upsampling head from the pre-trained VGGT point head. For each queried 3D point, we sample the 128-dimensional feature map from all valid source views and average the sampled features. We additionally encode its Euclidean ray depth relative to each source camera using a two-layer MLP and a ReLU activation. The resulting 32-dimensional embeddings are averaged over valid views and concatenated with the sampled image feature, yielding a 160-dimensional local spatial feature. The geometry decoder is an eight-layer MLP with a hidden dimension of 256. It maps the 160-dimensional local spatial feature to one SDF value and a 256-dimensional appearance feature. The appearance

decoder takes the appearance feature, the SDF gradient, and a positional encoding of the viewing direction. It is a two-layer MLP that predicts view-dependent RGB values.

Optimization. We use the AdamW optimizer (Loshchilov & Hutter, 2017) and the learning rate warms up to a maximum value and decreases according to a cosine schedule. During the first stage, the transformer-based backbone and DPT-style upsampling head use a peak learning rate of 1e-5, whereas the newly introduced geometry and appearance decoders use a peak learning rate of 2e-4. In the second stage, we reduce the latter to 1e-4.

Training Details. Each training sequence contains 16 views, including 12 context views and 4 novel views. We randomly sample 4 context views and use both novel views, resulting in 8 rendered views per iteration during the first training stage. We sample 1024 rays from each rendered view and apply the error-bounded sampling strategy (Yariv et al., 2021) to sample points along each ray. During the second training stage, we randomly sample 2 context views and 2 novel views to sample the rendered rays. We additionally sample 60,000 valid surface points from each rendered view to supervise the SDF and color predictions using $\mathcal{L}_{\text{point}}$. We set λ_1, λ_2, λ_3, λ_4, λ_5, λ_6, λ_7 to 0.1, 0.05, 1.0, 0.01, 0.01, 1.0, 0.1, respectively. We train the model on 4 NVIDIA A800 GPUs for 7 days.

B.2 Dataset Details

We train our IVGT using a diverse collection of datasets: ARKitScenes (Baruch et al., 2021), CO3Dv2 (Reizenstein et al., 2021), HyperSim (Roberts et al., 2021), MegaDepth (Li & Snavely, 2018), MVS-Synth (Huang et al., 2018), OmniObject3D (Wu et al., 2023), ScanNet (Dai et al., 2017), ScanNet++ (Yeshwanth et al., 2023), TartanAir (Wang et al., 2020), Unreal4K (Tosi et al., 2021) and WildRGBD (Xia et al., 2024). These datasets cover objects and scenes, including both real and synthetic data. We only require camera poses and RGB-D data as supervision (surface normals are obtained by predicting from RGB images using DSine (Bae & Davison, 2024)), which allows us to fully leverage existing 3D datasets. Across the two training stages, we dynamically adjust the sampling ratios of training sequences from different datasets.

B.3 Evaluation Details

We provide additional evaluation details for the results reported in Table 1. When reproducing the baselines, we resize the input images to the training resolution used by each respective method. SimpleRecon (Sayed et al., 2022), FineRecon (Stier et al., 2023), and MeshSplat (Chang et al., 2025) require camera poses as input. To ensure a fair comparison, we provide these methods with the camera poses estimated by our model. MeshSplat natively reconstructs a scene from an image pair. For multi-view inputs, we form pairs of adjacent views with substantial overlap, run MeshSplat independently for each pair, and fuse the resulting reconstructions to obtain the final mesh. For mesh evaluation, we first align the scale using the mesh vertices, uniformly sample points from the predicted and ground-truth meshes with the same sampling density, and then compute the evaluation metrics. The inference times reported in Table 1 are averaged over all test scenes and measured on a single NVIDIA A800 GPU. It is worth noting that for per-scene optimization methods, we report the average optimization time across all test scenes.

C More Visualization Results

We use additional scenes and objects from diverse datasets to demonstrate the quality of the colored meshes produced by our method. As shown in Figure 9, our method generalizes well across a variety of data, producing geometrically complete and visually consistent colored meshes without any test-time optimization.

D Limitations

Despite the promising results, IVGT has several limitations for future exploration:

- While our SDF-based rendering produces geometrically coherent outputs, the rendering quality in terms of high-frequency appearance details lags behind dedicated novel view synthesis methods



Figure 9: **Colored mesh reconstruction on diverse scenes and objects.**

such as those based on 3D Gaussian splatting. The smoothness prior enforced by Eikonal regularization, while beneficial for surface coherence, suppresses fine-grained geometric structures such as thin objects and sharp edges.

- The current framework assumes a static scene and a bounded spatial extent, limiting its applicability to dynamic environments or large-scale unbounded outdoor scenes.
- The continuous 3D query mechanism requires projecting each sampled point onto all input views and aggregating features, which is more computationally intensive than direct pixel-aligned decoding at inference time, which may hinder real-time deployment.

Addressing these (e.g., incorporating appearance-oriented decoding heads, handling dynamic content, or developing more efficient query strategies) remains an important direction for future research.