DG-Recon: Depth-Guided Neural 3D Scene Reconstruction

Jihong Ju Ching-Wei Tseng Oleksandr Bailo Georgi Dikov Mohsen Ghafoorian
XR Labs, Qualcomm Technologies, Inc.
{jihoju, chintsen, obailo, gdikov, mghafoor}@qti.qualcomm.com

Abstract

A key challenge in neural 3D scene reconstruction from monocular images is to fuse features back projected from various views without any depth or occlusion information. We address this by leveraging monocular depth priors, which effectively guide the fusion to improve surface prediction and skip over irrelevant, ambiguous, or occluded features. Furthermore, we revisit the average-based fusion used by most neural 3D reconstruction methods and propose two alternatives, a variance-based and a cross-attention-based fusion module, that are more efficient and effective than the average-based and self-attention-based counterparts. Compared to the NeuralRecon baseline, the proposed DG-Recon models significantly improve the reconstruction quality and completeness while remaining in real-time. Our method achieves state-of-the-art online reconstruction results on the ScanNet dataset and is on par with the current best offline method, which repeatedly accesses keyframes from the entire video sequence. Our ScanNet-trained model also generalizes robustly to the challenging 7-Scenes dataset and a subset of SUN3D containing scenes as big as an entire floor.

1. Introduction

Reconstruction of 3D scenes is a fundamental problem in 3D perception of environments, constituting a crucial component of various application domains ranging from robotics and autonomous vehicles to augmented and virtual reality. For instance, in the augmented/virtual reality use case, not only the accuracy of the reconstructed meshes but also the runtime efficiency is important in enabling real-time safe user navigation, successful occlusion rendering, and plausible physical simulations on edge devices.

Most traditional 3D scene reconstruction pipelines consist of dense depth prediction and a multi-view depth integration process [31, 6] to create truncated signed distance function (TSDF) as a geometrical representation that enables mesh extraction using the marching cubes algorithm [26]. While such processes are simple and intuitive, the non-learnable fusion part is unable to effectively incorporate higher-level inductive biases to handle inconsistent and/or noisy depth estimations from different views. On the other hand, a newly emerging category of methods [30, 42, 1, 41] aims at learning to directly predict the TSDF by back projecting representations from posed images and then fusing them into volumetric representations of the underlying scenes. These neural methods are in practice either found to be suffering from lower accuracy and incomplete geometries [42], or too costly for real-world and real-time use cases on edge-devices [30, 1, 41].

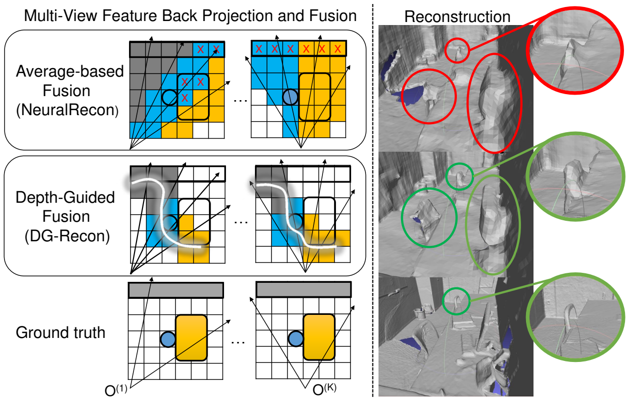
Multi-View Feature Back Projection and Fusion
Reconstruction
Average-based Fusion (NeuralRecon)
Depth-Guided Fusion (DG-Recon)
Ground truth
...
O(1)
O(K)

Figure 1. **Depth-guided back projection and fusion**. Each grid on the left figure represents a digitized 2D world. The orange rounded rectangle represents a table. The blue circle sketches a chair and the grey rectangle indicates the wall. Non-white cells picture back-projected features from K different camera views $\mathbf{O}^{(1)}, \ldots, \mathbf{O}^{(K)}$. A cell with a red cross in it indicates an erroneous back projection. The depth priors, denoted as the white surface curve in the middle row, introduce geometry awareness to the back projection and cross-view fusion. Objects, *e.g*. chairs and the monitor, become sharper, more complete, and better separated.

Such undesirable properties can be attributed to bottlenecks in important components, *i.e*. feature back projection, feature fusion, and occupancy prediction. More specifically, existing neural reconstruction methods back project image features all along the rays into the volumetric representations resulting in 1) non-sparse representations and 2) potential erroneous feature association on the occluded objects. Besides, the feature fusion is either based on simple averaging [42, 30] that is ineffective in properly modeling the multi-view consensus or is based on the self-attention mechanism [1] that is inefficient as it scales quadratically with the number of views.

In this work, we build our method upon the most scalable algorithm of the 3D volumetric reconstruction category, NeuralRecon [42], and revisit feature back projection and fusion with the help of depth priors. The depth-guided back projection reduces erroneous feature associations with occluded objects and introduces sparse representations even before fusion, as illustrated in Figure 1. Early availability of sparsity enables the choice of more expressive representation aggregation schemes without significant computational costs. The proposed variance- and cross-attention-based fusions are both more effective than the average-based fusion and more efficient than the self-attention-based fusion. Finally, the depth prior also helps improve the reconstruction completeness over the baseline [42] by replacing the overfitted occupancy prediction with the depth-derived occupancy mapping. To summarize, our major contributions are:

- We propose to integrate depth priors to the feature back projection and occupancy prediction component in 3D volumetric scene reconstruction methods, which improves cross-view feature association and creates sparse volumetric representation before fusion.
- We formulate and propose two simple and scalable surrogate feature fusion schemes, the variance and cross-attention, that are shown to be effective and efficient as compared to the formulations commonly used.
- Our comprehensive empirical evaluations on ScanNet [5], 7-Scenes [17] and SUN3D [50] show that our proposed method is the new state-of-the-art in 3D scene reconstruction considering the accuracy-efficiency tradeoff.

2. Related Work

Monocular depth estimation. The pioneering work of Saxena *et al*. [35] estimates dense depth from a single input image using local feature extractors and Markov random fields. Subsequent works [12, 11, 25, 24, 14, 19] leverage convolutional neural networks (CNN) to substantially increase the accuracy. More recently, [32] adapts the vision transformer (ViT) architecture [8] to generic dense prediction tasks and [55] extends it to unsupervised depth estimation. However, despite the superior prediction quality, [49] suggests that heavy ViT backbones are not practical for real-time applications and instead opt for a convolutional encoder-decoder network. Since real-time performance is essential to our work, we adopt the convolutional backbone.

Multi-View Stereo networks. Multi-View Stereo (MVS) methods estimate depth for the reference frame using one or more source viewpoints. Recent works [44, 22, 9, 23] extend the classical MVS methods [39, 15, 38] with the learning-based techniques to construct 3D cost volumes from multi-view representations and to regress the dense depth map. [36] proposes to reduce the depth dimension of the 3D cost volumes with parallel multi-layer perceptron (MLP) which allows integrating multi-view information without 3D convolution. Overall, deep MVS methods produce more accurate and consistent depth than monocular depth models. However, depth observations are sensitive to occlusion and non-learnable TSDF fusion methods [31, 56] lack the reasoning capability in 3D which is crucial to complete the occluded missing geometry.

VolumeFusion [4] combines the deep MVS method with volumetric TSDF prediction via pose-invariant 3D Conv. [33] iteratively refines 2D depth and 3D feature clouds which improves the reconstruction quality but takes tremendous time due to its cyclic refinement nature. Both methods require the entire sequence to be available before processing and are not feasible for online reconstruction in real-time.

Neural 3D Reconstruction. Atlas [30] proposes direct TSDF volume prediction from back-projected image features with a 3D CNN. But its volumetric prediction on the entire scene limits update frequency and scalability to large scenes. NeuralRecon [42] addresses the efficiency and scalability issue by adopting a sparse 3D CNN for TSDF prediction only in local fragments spanning the view frustum of a few neighboring frames. The local TSDF prediction is integrated into the global volume by direct replacement. Our method is built on top of the NeuralRecon [42] and addresses its issue of incomplete reconstruction and over-smoothed object shapes with the help of depth guidance and improved fusion mechanisms.

TransformerFusion [1] and VoRTX [41] propose to utilize the attention mechanism for multi-view feature fusion. But their self-attention-based fusion modules scale quadratically with the number of views. The inefficiency of self-attention causes their methods to either not meet the real-time requirement [1] or even execute offline only [41]. Our DG-Recon, on the other hand, runs in real-time and makes updates every 9 frames thanks to the linear scalability of the variance- and cross-attention-based fusion modules.

Online depth fusion methods [46, 47] deliver frequent 3D reconstruction in real-time but require dedicated depth sensors for accurate depth information. Our method however relies only on a rough estimation of the depth from monocular depth models.

Neural implicit representation. NeRF [28] is another approach toward high-fidelity 3D scene representation. It overfits an MLP to predict the density and radiance given a 3D position and the viewing angle. High-fidelity novel

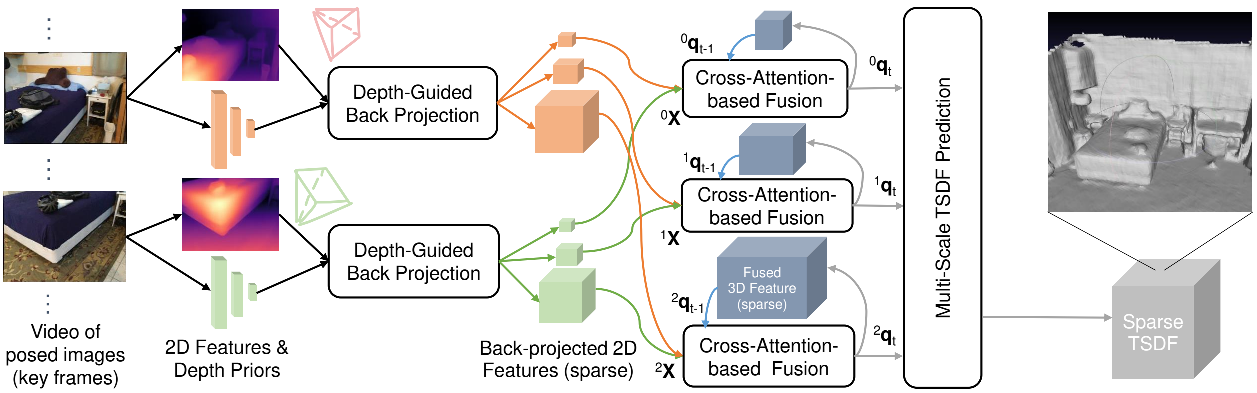
Video of posed images (key frames)
2D Features & Depth Priors
Depth-Guided Back Projection
Depth-Guided Back Projection
Back-projected 2D Features (sparse)
${}^0\mathbf{q}_{t-1}$
${}^0\mathbf{X}$
${}^0\mathbf{q}_t$
Cross-Attention-based Fusion
${}^1\mathbf{q}_{t-1}$
${}^1\mathbf{X}$
${}^1\mathbf{q}_t$
Cross-Attention-based Fusion
Fused 3D Feature (sparse)
${}^2\mathbf{q}_{t-1}$
${}^2\mathbf{X}$
${}^2\mathbf{q}_t$
Cross-Attention-based Fusion
Multi-Scale TSDF Prediction
Sparse TSDF

Figure 2. **DG-Recon overview**. Given K consecutive keyframes, DG-Recon first estimates depth and extracts representations from RGB images. The extracted features are then back projected to a sparse volume using the estimated depth prior and camera pose. Given previous 3D features, the cross-attention-based fusion module recursively attends to the newly extracted features and updates the fused features. This depth-guided feature fusion operates at three different scales and provides the inputs to the multi-scale local TSDF prediction based on sparse 3D Conv and GRU fusion similar to NeuralRecon [42] for global TSDF update. ${}^i\mathbf{X}$ and ${}^i\mathbf{q}$ denote features on the i-th scale.

views can be generated from this neural implicit representation based on volume rendering. Recent works introduced similar depth priors as ours to NeRF for better view synthesis [7, 34] and for better depth estimation [48] purposes but not for better 3D reconstruction. [45, 52] replaced the density prediction in NeRF with signed distance function (SDF) which improves the NeRF-based surface reconstruction. ManhattanSDF [20] further enables scene reconstruction of an entire room by jointly optimizing the rendered geometry and semantics. However, these methods require overfitting to the target scene and do not generalize to novel scenes. [2, 54] improve generalization by conditioning the MLP additionally on the learnable warped representations but their methods focus on view synthesis rather than reconstruction. Our method, on the other hand, can reconstruct unseen environments without any fine-tuning.

Concurrent works. Recent works also identify the feature back projection issue in parallel and they address it in various ways. Inspired by MVS methods, CVRecon [13] proposes to integrate view-dependent information from cost volumes. FineRecon [40] instead employs MVS depth [36] directly. Both works build upon the offline method VoRTX [41] which is not suitable for real-time incremental reconstruction. Online method, VisFusion [16], tackles the unprojection ambiguity with pair-wise feature similarity which scales quadratically with the number of views while ours remains linear. Zuo *et al*. [58] rely on compute-heavy MVS networks for depth priors while DG-Recon utilizes a lightweight monocular depth network.

3. Method

Given a stream of keyframes $\{\mathbf{I}^{(k)} \in \mathbb{R}^{H\times W\times 3}\}_{k=1}^{N}$, selected from a monocular video sequence, and the corresponding camera poses $\{\mathbf{T}^{(k)} \in SE(3)\}_{k=1}^{N}$ obtained from an online 6DoF localization system [6, 37], our DG-Recon incrementally updates the sparse 3D features and sparse TSDF representing surfaces in a scene. Following [42], the incremental updates are kept local to the frustum of 9 consecutive keyframes to deal with large scenes. An overview of DG-Recon is demonstrated in Figure 2.

3.1. Depth-guided back projection and fusion

DG-Recon incorporates the depth priors estimated from monocular images 1) to guide the feature back projection from perspective view to 3D space, 2) to serve the near-surface occupancy probability modeling for 3D volumes sparsification, and 3) to feed the cross-view fusion module with auxiliary features. This section describes these three components in separate paragraphs.

Feature back projection. Given a predicted depth map for the k-th keyframe $\mathbf{D}^{(k)} \in \mathbb{R}^{H\times W}$, DG-Recon back-projects 2D features $\mathbf{F}^{(k)} \in \mathbb{R}^{H\times W\times C}$ along the rays only to those voxels within a fixed distance Δ from the corresponding estimated depth surface, as shown in Figure 1,

$$\mathbf{f}_{ijk}^{(k)} = \begin{cases} \mathbf{F}_{uv}^{(k)}, & \left|z_{ijk}^{(k)} - \mathbf{D}_{uv}^{(k)}\right| < \Delta \\ \mathbf{0}, & \text{otherwise} \end{cases} \tag{1}$$

where ijk denotes a voxel in 3D space, uv the corresponding 2D pixel and z the depth projected from voxel ijk.

The depth-guided back projection not only prevents populating voxels with irrelevant 2D features, *e.g.* when an object is occluding another object but also introduces the sparsity earlier on in the pipeline. The availability of sparsity information before fusion enables more compute-heavy fusion modules without sacrificing much overall efficiency.

Occupancy mapping. The depth prior is also utilized in the 3D occupancy mapping as a replacement of the occupancy prediction heads in NeuralRecon [42]. More specifically, voxel's activation for TSDF prediction is derived from the depth priors using occupancy grid mapping [29]. Assuming a static world and independent voxels, the probability of a voxel being occupied, *i.e.* close to a surface, given depth observation from k different views $p(m_{ijk}|z_{ijk}^{(1:k)}) \in [0,1]$ is updated recursively by

$$l(m_{ijk}|z_{ijk}^{(1:k)}) = l(m_{ijk}|z_{ijk}^{(1:k-1)}) + l(m_{ijk}|z_{ijk}^{(k)}), \qquad (2)$$

where the log-odds notation $l(\cdot) = \log \frac{p(\cdot)}{1-p(\cdot)}$ is introduced for better efficiency and computation stability. The first term in Eq. 2 is the recursive term and the second term is derived from the occupancy probability given one depth estimation modeled by

$$p(m_{ijk}|z_{ijk}^{(k)}) = \frac{\mathcal{N}(z_{ijk}^{(k)}|\mu, \sigma^2)}{\mathcal{N}(\mu|\mu, \sigma^2)}, \qquad (3)$$

where the Gaussian distribution $\mathcal{N}$ centers at the corresponding monocular depth estimation $\mu = \mathbf{D}_{uv}^{(k)}$ with a fixed standard deviation $\sigma = \Delta$, the back projection margin.

Auxiliary geometry feature. In addition to the learned back-projected features, DG-Recon introduces geometrical representation as auxiliary inputs to the multi-view fusion module. Experiments in Section 5.4 show that the most effective view-dependent geometry features are the depth offset $\mathbf{d} \in \mathbb{R}$ and ray direction $\mathbf{r} \in \mathbb{R}^3$:

$$\begin{aligned} \mathbf{d}_{ijk}^{(k)} &= z_{ijk}^{(k)} - \mathbf{D}_{uv}^{(k)}, \\ \mathbf{r}_{ijk}^{(k)} &= \frac{\mathbf{p}_{ijk} - \mathbf{o}^{(k)}}{\left\| \mathbf{p}_{ijk} - \mathbf{o}^{(k)} \right\|}, \end{aligned} \qquad (4)$$

where $\mathbf{p}_{ijk}$ is the world coordinate of the voxel ijk and $\mathbf{o}^{(k)}$ is the camera center of the k-th keyframe. Concatenating these view-dependent geometrical representations to the back-projected 2D features $\mathbf{x}_{ijk}^{(k)} = \begin{bmatrix} \mathbf{f}_{ijk}^{(k)} & \mathbf{d}_{ijk}^{(k)} & \mathbf{r}_{ijk}^{(k)} \end{bmatrix}$, enables the fusion model to better weigh the importance of features from different views. For simplicity, we omit the ijk subscription for the rest of the paper.

3.2. Revisiting multi-view feature fusion

Averaging [42, 30] back-projected features cross-views is efficient but sometimes leads to over-smoothed geometry. Self-attention-based [1, 41] fusion is more expressive but introduces significant computational overhead. DG-Recon provides two alternative fusion options that are both efficient and effective.

Variance-based fusion. Given features extracted from K different views, $\mathbf{X} = \begin{bmatrix} \mathbf{x}^{(1)} & \dots & \mathbf{x}^{(K)} \end{bmatrix}$, the variance module computes the estimated variance of the multi-view features as the fused 3D feature.

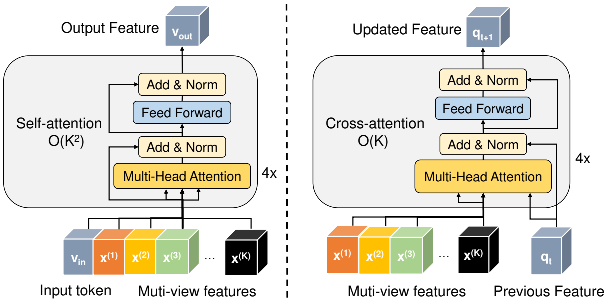

Figure 3. **Comparison between self-attention-based and cross-attention-based fusion of multi-view features**. $\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(K)}$ are the back-projected features from K views. $\mathbf{v}$ is the input/output token for the self-attention module and $\mathbf{q}$ denotes the recursively updated 3D feature for the cross-attention-based fusion.

$$\mathbf{v} = \mathrm{var}(\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(K)}) \qquad (5)$$

The variance propagates information about view differences which better represents the geometry than average.

Cross-attention-based fusion. For even more expressiveness and less computational cost than [1, 41], DG-Recon offers the cross-attention-based fusion. It exploits not only features extracted from K recent views but also the previous fusion outputs. The multi-head attention module [43] is fed with the query $\mathbf{Q}$, key $\mathbf{K}$ and value $\mathbf{V}$ tokens defined by:

$$\begin{aligned} \mathbf{Q}_t &= \mathbf{q}_{t-1} \in \mathbb{R}^{d \times 1}, \\ \mathbf{K}_t, \mathbf{V}_t &= \mathbf{X} \in \mathbb{R}^{d \times K}, \end{aligned} \qquad (6)$$

where $\mathbf{q}_{t-1}$ is the previously fused features for a 3D voxel, $\mathbf{X}$ is the stack of newly extracted features from K recent views at timestamp t, and d is the feature dimension. As shown in Figure 3, the recursively updated 3D feature for a voxel is computed by 4 multi-head cross-attention layers:

$$\mathbf{q}_t = \mathrm{MultiHeadAttention}_{\times 4}(\mathbf{q}_{t-1}, \mathbf{X}, \mathbf{X}), \qquad (7)$$

which again becomes the query token for the next feature update step at $t+1$. If a voxel has never been observed before, a learnable initial query vector $\mathbf{q}_0$ is used instead.

Because the time and space complexity of the cross-attention-based fusion module grows linearly with the number of views O(K), it is more efficient than O(K^2) self-attention-based fusion [1], as shown in the supplementary Section 4.2, while being more effective than the non-learnable average or variance-based fusion as backed by the experiments in Section 5.4.

3.3. Depth prior from monocular depth estimation

The depth prior $\mathbf{D}$ for each keyframe may come from various sources, *e.g.* monocular or multi-view depth estimation or even dedicated depth sensors. For a better trade-off between accuracy, feasibility, and computational cost, we adopt the monocular depth estimation approach with a network consisting of a ResNet34 [21] encoder and the convolutional decoder from [32] as proposed in [49]. Due to its monocular nature, this model is unable to predict the absolute metric depth and thus suffers from inconsistent predictions. To resolve this, inspired by [3, 51, 27], we provide additionally sparse metric depth for keypoints, which are extracted from a 6DOF visual odometry algorithm. The depth prior network can be trained separately in a fully-supervised [12, 11] or self-supervised manner [18, 57, 19].

3.4. Implementation details

DG-Recon follows the general implementation of NeuralRecon [42]. The occupancy prediction head is removed because the occupancy in DG-Recon is directly estimated from the depth priors. The back projection margin Δ is set empirically to 8 voxels distance and the number of views K in a local fragment is set to 9 following [42]. The training target for the TSDF head is derived from the entire ground truth depth sequence instead of the accumulated frames so far. This modification encourages the network to complete the not yet observed geometry. To prevent punishing the network from completing the occluded geometry, a global visibility mask, derived from depth sensor readings, was applied when calculating the TSDF loss. During training, lower-level voxels were upsampled to higher levels only if the TSDF prediction is between [-0.9, 0.9]. This configuration helps the training focus on difficult samples. Unlike NeuralRecon [42]'s two-stage training strategy, DG-Recon was trained in one stage with GRU fusion activated.

The sparse depth inputs to the standalone DPT [32] depth model were obtained by following the tutorial of COLMAP [37] for sparse reconstruction from known camera poses. The depth network was trained fully-supervised with the depth sensor readings as the training target. An element-wise BerHu loss [24] was adopted and the pixels missing ground truth depth were masked out in the loss.

More implementation details including the network architectures and hyperparameters for training are attached in Section 2 of the supplementary materials.

4. Experimental Setup

Datasets. ScanNet [5] consists of 1613 RGB-D scans of indoor scenes, of which 1201 scans were used for training, 312 for validation, and 100 for testing. The official cleaned meshes were used as ground truth in the evaluation. To avoid exhausting the test with different method variations, We used the validation set for ablation studies while keeping the test set for comparison to the state-of-the-art methods only. For generalization evaluation, 13 RGB-D scans from 7-Scenes [17] selected by [9] covering all 7 scenes were used for testing. 23 RGB-D sequences from SUN3D [50] preprocessed by [53], containing different indoor environments ranging from one room to an entire floor, were also selected for testing. The ground truth meshes for 7-Scenes and SUN3D were created from the depth sensor readings using the TSDF fusion script provided by [30].

Baselines. NeuralRecon [42], a real-time online volumetric 3D reconstruction method, is the direct baseline to compare DG-Recon against. Atlas [30] and TransformerFusion [1] support online updating of the fused features but are not optimized for real-time reconstruction for large scenes. 3DVNet [33] and VoRTX [41] further require the entire video sequence to be available. While SimpleRecon [36], a deep MVS method, doesn't belong to the category of volumetric reconstruction methods, we find it useful as another SOTA model to compare our model against.

Reconstruction metrics. We adopted the evaluation protocol established by [1] to compare the reconstructed 3D mesh against the ground-truth mesh. Please refer to Section 3 of the supplementary material for detailed metric definitions. To obtain meshes from the TSDF prediction, we run marching cubes [26] at zero level set. A visibility mask was applied to the predicted mesh following [1] to reduce the impact of ground-truth mesh incompleteness due to missing and noisy depth sensor readings. Points were sampled uniformly from the meshes to account for the resolution difference between the ground truth and predicted meshes.

Efficiency metrics. *Online/offline* categorizes if the method requires access to all frames in a video sequence during reconstruction. Online methods can process frame-by-frame sequentially and update the scene representation incrementally whilst offline methods cannot. *Frame Per Second (FPS)* measures the frames processed per second, assuming the reconstruction is updated every 9 keyframes. All FPS, except for [1], was measured on a single 11GB NVIDIA 2080Ti using scene0707_00 from ScanNet [5].

Depth rendering metrics. We adopted the evaluation pipeline from NeuralRecon [42] to evaluate the accuracy of the rendered depth images. The unknown depths from the ground-truth depth image are excluded from depth evaluation. The missing depths from the rendered image are measured and compared by *Comp2D*. Detailed metrics definitions can be found in Section 3 of the supplementary.

5. Experimental Results

5.1. Evaluation of 3D reconstruction on ScanNet

3D reconstruction performance of DG-Recon was evaluated and compared to SOTA methods for both efficiency and accuracy on the ScanNet test set. Figure 4 illustrates the trade-off between reconstruction F-score and computation efficiency (FPS). DG-Recon, with the cross-attention-based fusion (c-att), outperforms the other online methods

Method	**Online**	**FPS↑**	**Acc. ↓**	**Comp.↓**	**Chamfer↓**	**Precision↑**	**Recall↑**	**F-score↑**
3DVNet[33]	x	0.4	6.73	7.73	7.22	0.655	0.596	0.621
VoRTX[41]	x	2	4.31	7.23	5.77	0.767	0.651	0.703
Atlas [30]	✓	10	7.16	7.61	7.38	0.675	0.605	0.636
TransformerFusion[1]	✓	7^*	5.52	8.27	6.89	0.728	0.600	0.655
SimpleRecon[36]	✓	16	5.53	6.09	5.81	0.686	0.658	0.671
NeuralRecon[42]	✓	46	5.09	9.13	7.11	0.630	0.612	0.619
DG-Recon (c-att)	✓	20	3.94	6.82	5.38	0.769	0.636	0.694
DG-Recon (var)	✓	34	4.40	6.49	5.44	0.732	0.628	0.674
Depth priors-only	✓	133	8.49	7.15	7.82	0.607	0.561	0.580

Table 1. **Evaluation of the 3D mesh on the ScanNet test set**. Reconstruction results for previous works were taken from [36] following the evaluation pipeline of [1]. Frames per second (FPS) was measured based on the per-frame time and per-update time amortized over 9 keyframes. This is different from the FPS reported by offline methods, which runs TSDF prediction only once for the entire video sequence. *FPS was measured at chunk size $(1.5\text{m})^3$ for [1] and was measured at $(5.12\text{m})^3$ for DG-Recon. Red cells mark the best number, orange the second best, and yellow the third best. The depth priors-only model was listed for ablation study and not ranked. Further comparisons to the concurrent works can be found in the supplementary materials Section 4.

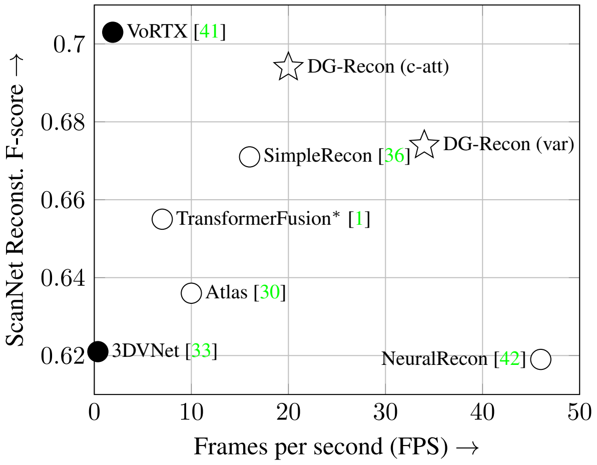

Figure 4. **The accuracy-efficiency tradeoff for 3D reconstruction methods**. The closed circles denote offline methods while the open circles and stars (ours) represent online methods. (c-att) represents cross-attention-based fusion and (var) corresponds to variance-based fusion. * FPS was measured by the authors [1] with a chunk size 43x smaller than ours.

in F-score. It reaches 20 keyframes per second assuming an update frequency of every 9 keyframes. The F-score is even on par with the best offline method, VoRTX [41], which requires access to the entire video sequence and runs significantly slower. The variance-based DG-Recon (var) achieves a comparable F-score as SimpleRecon [36] while being 2x faster.

Table 1 lists the detailed reconstruction results for SOTA scene reconstruction methods. Both DG-Recon variants appear in the top three for most metrics. Compared to the NeuralRecon [42] baseline, DG-Recon improves its F-score from 0.619 to 0.674 with the variance-based fusion and to 0.694 with cross-attention-based fusion. The corresponding FPS drops by 12 for DG-Recon (var) and 26 for DG-Recon (c-att) due to the standalone depth model and additionally the learnable cross-attention-based fusion. Both models still run at higher FPS and score higher for reconstruction than Atlas [30] and especially TransformerFusion [1] which relies on the less efficient self-attention-based fusion. We also present the reconstruction results of a non-learnable TSDF fusion [56] using the depth priors predicted by the monocular depth network. It proves that the learnable feature extraction, fusion, and TSDF prediction are crucial to the performance of DG-Recon.

Figure 5 compares the reconstruction of DG-Recon against SOTA volumetric reconstruction methods. Overall, DG-Recon delivers objects with sharper shapes, *e.g*. the sink, toilet (row 1), kitchen worktops (row 2), and nightstands (row 3), than the other methods. The chairs in row 5 are well separable whilst they almost look like a single bench with Atlas [30] and VoRTX [41]. Compared to its baseline [42], DG-Recon produces more complete meshes (row 4, 6, and 7). It completes the invisible corners which is an ability lacking in [42] because the occupancy prediction overfits the incomplete ground truth.

We further compare qualitatively the reconstruction results of DG-Recon against SimpleRecon [36], the SOTA MVS method, in Figure 6. Due to the 3D volume resolution limitation, DG-Recon's reconstruction is of slightly lower fidelity than SimpleRecon. But DG-Recon learns to de-noise floaters and fill in occluded corners, a missing capability by SimpleRecon due to the lack of 3D reasoning.

5.2. Generalization to other datasets

The off-the-shelf transferability of DG-Recon to other reconstruction datasets without fine-tuning is evaluated on 7-Scenes [17] and SUN3D [50]. Both datasets were cap-

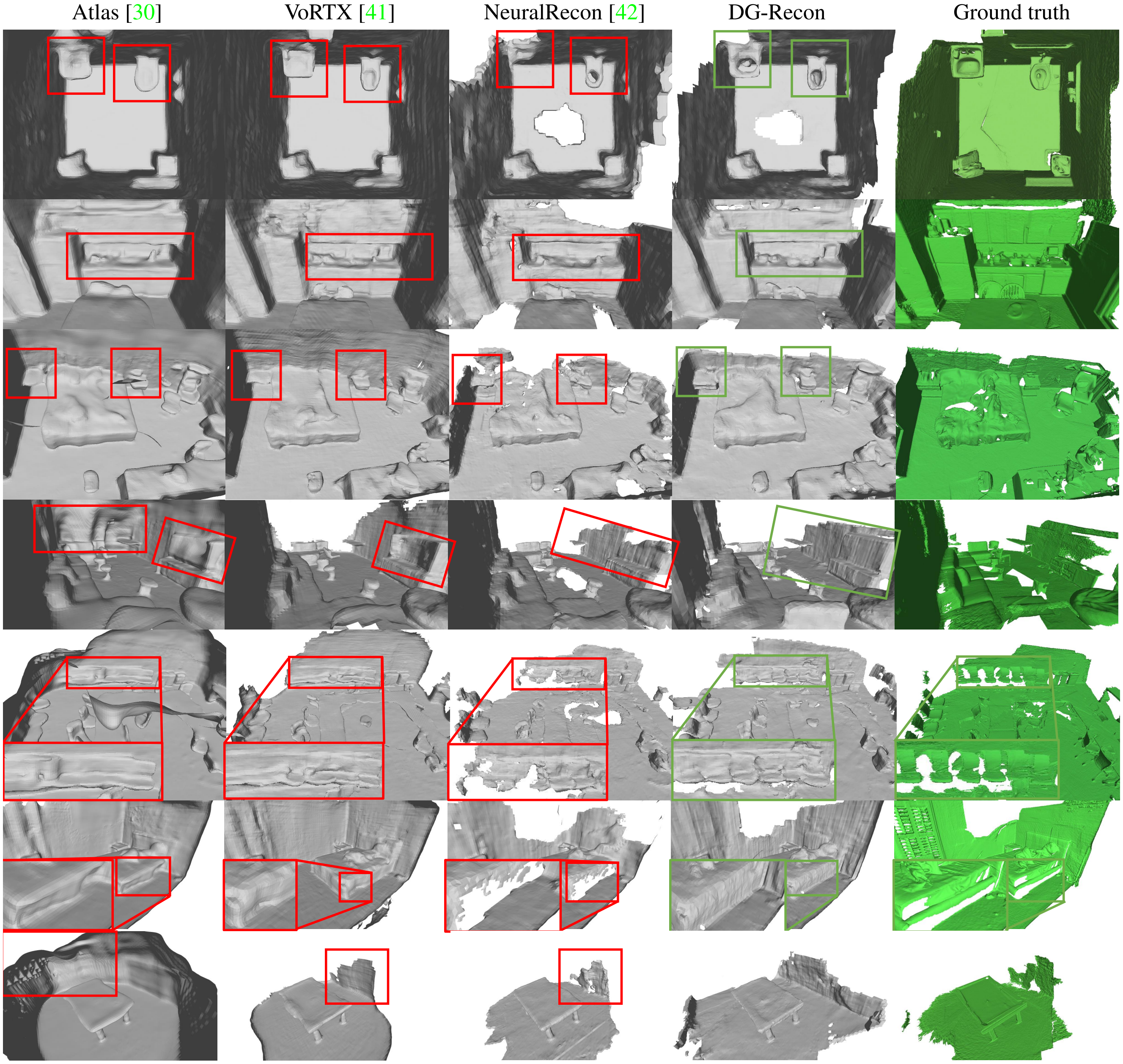

Figure 5. **Qualitative analysis of the volumetric 3D reconstruction methods on ScanNet**. Reconstructions for NeuralRecon [42] and Atlas [30] were reproduced with the official codebase and published weights. Meshes for VoRTX [41] were provided by the original authors. To produce these meshes, Atlas and VoRTX run inference for the entire scene once at the end whilst the rest make continuous updates and dump the final state. Reconstruction errors are highlighted by red boxes and quality shapes are marked by green boxes. More qualitative comparison is available in the supplementary material Section 5.

tured with Kinect V1, a different camera setup from ScanNet [5]. Table 2 shows that DG-Recon consistently improves the reconstruction F-score of the NeuralRecon [42] baseline by 18% and 11% on 7-Scenes and SUN3D respectively. Moreover, it achieves a better balance between precision and recall and outputs much more complete reconstructions for both datasets. Atlas [30] on SUN3D suffers from its limitation of processing the entire scene in one go. The SUN3D testing scenes can span over 25 meters, which doesn't fit into the GPU memory for Atlas [30]. Its recall is therefore lower and the completeness error is one order of magnitude higher than the other two methods. Qualitative analysis for both datasets can be found in Section 5 of the supplementary materials.

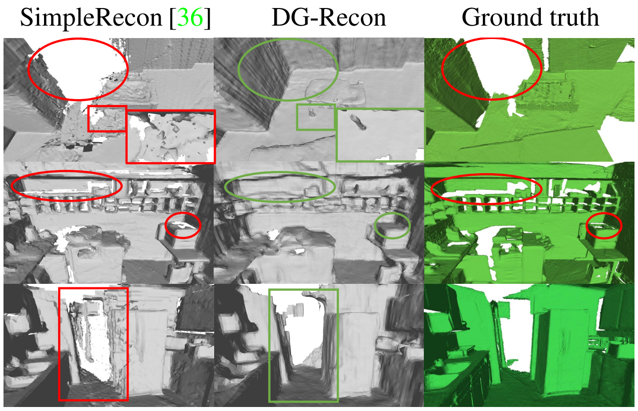

Figure 6. **Qualitative comparison with SimpleRecon on ScanNet**. The red circles highlight missing geometries for SimpleRecon [36] and for ground truth. The green circles showcase DG-Recon's capability of completing occluded geometry. The red and green boxes mark noisy floaters and clean geometry respectively.

Dataset	**Method**	**Acc ↓**	**Compl ↓**	**Prec ↑**	**Rec ↑**	**F-score ↑**
7-Scenes	Atlas[30]	8.6	10.8	0.496	0.460	0.477
	NeuralRecon[42]	**5.7**	16.9	**0.582**	0.383	0.459
	DG-Recon (c-att)	6.2	**8.5**	0.566	**0.522**	**0.542**
SUN3D	Atlas[30]	8.3	83.9*	0.443	0.329	0.366
	NeuralRecon[42]	**6.8**	16.2	**0.520**	0.338	0.408
	DG-Recon (c-att)	7.7	**9.6**	0.473	**0.434**	**0.451**

Table 2. **Reconstruction generalization to the 7-Scenes [17] and SUN3D [50] datasets**. **Bold** denotes the best number and underline the second best. [30] and [42] were evaluated using the official implementation and pre-trained weights. * High completeness error caused by the fact that some testing scenes do not fit into the GPU memory for [30].

5.3. Evaluation of depth rendering

Method	**Abs Rel ↓**	**Abs Diff ↓**	**Sq Rel ↓**	**$\delta < 1.25$ ↑**	**Comp2D ↑**
Atlas[30]	0.064	0.118	0.042	92.3	0.963
NeuralRecon[42]	0.065	**0.098**	**0.038**	93.3	0.891
SimpleRecon[36]	**0.061**	0.103	0.041	**94.1**	**0.969**
DG-Recon (c-att)	0.062	0.099	0.040	93.7	0.966

Table 3. **Evaluation of rendered depth on the ScanNet test set**. **Bold** denotes the best number and underline the second best. Both accuracy (columns 2-5) and completeness (column 6) matter for the rendering applications. Note that [36] reported lower errors because the rendered depth, different from the direct prediction, is subject to the 3D reconstruction quality and camera pose errors.

We evaluate depths rendered from the reconstructed scenes on the ScanNet test set following the same setup as [42]. The errors and completeness of the rendered depth are listed in Table 3 for [30, 42, 36] and DG-Recon. DG-Recon achieves comparable accuracy as NeuralRecon [42] while improves the depth completeness from 89.1% to 96.9%. The overall quality of the rendered depth is better than Atlas [30] and on par with SimpleRecon [36]. Qualitative analysis of the rendered depth can be found in the supplementary material Section 5.

5.4. Ablation studies of individual components

In this section, we report the results of the ablation studies for three main contributions: 1) depth guidance in feature fusion, 2) different fusion mechanisms, and 3) auxiliary geometry features.

Given that the validation set comprises only 142 unique scenes, a subset of the ScanNet validation containing 142 scans was used for ablation studies. We found empirically that this one-scan-per-scene subset is equally representative as the full set. To prevent punishing geometry completeness, visibility masks were created for the validation set following [1] and applied in the ablation studies.

Depth guid.	**Feat. Fuse**	**Acc ↓**	**Compl ↓**	**Prec ↑**	**Rec ↑**	**F-score ↑**
	avg(**f**)	**4.7**	10.7	**0.716**	0.507	0.590
✓	avg(**f**)	7.8	6.4	0.596	0.625	0.608
✓	avg(**f,d**)	7.6	6.6	0.603	0.618	0.608
✓	avg(**f,d,r**)	8.2	6.3	0.582	0.618	0.597
	c-att(**f**)	7.0	8.9	0.581	0.497	0.533
✓	c-att(**f**)	8.3	**6.2**	0.586	0.625	0.602
✓	c-att(**f,d**)	7.3	6.4	0.631	0.643	0.634
✓	c-att(**f,d,r**)	7.2	**6.2**	0.633	**0.653**	**0.640**
✓	c-att(**d,r**)	7.4	6.9	0.597	0.598	0.595

Table 4. **Ablation study of depth guidance and auxiliary geometry features**. avg($\cdot$) represents feature average and c-att($\cdot$) corresponds to the cross-attention-based fusion module. **f** denotes the back-projected 2D features. **d** is the offset between the depth prior and the projected depth. $\mathbf{r} \in R^3$ is the unit vector pointing from the 3D point of interest to the camera center of each view.

Depth guidance. The influence of depth guidance in DG-Recon for feature back projection and fusion is ablated and reported in Table 4. Starting from the NeuralRecon [42] baseline (first row in Table 4), adding the depth-guided back projection and occupancy mapping improves the F-score from 0.590 to 0.608. The difference is even bigger, 0.533 *vs.* 0.608, when the average is replaced by cross-attention in the fusion module. Moreover, depth guidance helps DG-Recon simplify NeuralRecon [42]'s two-stage training strategy to a one-stage strategy. Switching from non-learnable average-based fusion to learnable cross-attention-based fusion becomes possible without hyperparameter tuning. Overall, the depth guidance allows DG-Recon to balance precision and recall. The reconstructed mesh becomes more complete while still being accurate.

Auxiliary geometry features. Table 4 compares different input feature combinations for the cross-attention fusion module. Specifically, providing the distance offset to the depth priors **d** in addition to the back-projected features **f**, F-score increases 5% to 0.634. Adding the viewing angle **r** further improves the F-score to 0.640. The auxiliary geometry features are not as beneficial when the baseline fusion method average is in use. The additional viewing angle even slightly diminishes the F-score. With a learnable cross-attention-based fusion mechanism, DG-Recon shows the capability of better utilizing auxiliary information.

Furthermore, relying on the geometry features without image features results in an F-score of 0.595 only, 7% below the full set of features. It indicates that both the image-extracted features and the geometry features are key contributors to the reconstruction improvement.

Fusion	FPS ↑	Acc ↓	Compl ↓	Prec ↑	Recall ↑	F-score ↑
average	34	8.2	6.3	0.582	0.618	0.597
variance	34	7.7	6.4	0.628	0.636	0.629
cross-attention	20	7.8	**5.9**	0.610	**0.658**	0.631
+learnable query	20	**7.2**	6.2	**0.633**	0.653	**0.640**

Table 5. **Ablation study of different fusion mechanisms**. Features from multiple views, including back-projected image representations and the auxiliary geometry features, were fused with the different fusion operations.

Fusion mechanisms. We also experiment with various fusion mechanisms, average, variance, and cross-attention with a fixed initial query or with a learnable initial query. Table 5 shows that the F-score is boosted significantly without FPS dropping by simply replacing the average of multi-view features with variance. The introduction of the cross-attention-based fusion module further raises the validation F-score to 0.631 with a fixed initial query vector and to 0.640 with a learnable initial query. While being more accurate, the cross-attention module introduces a computational overhead of 14 FPS less than the non-learnable variants.

5.5. Effect of depth prior quality

Depth Prior	Acc↓	Compl↓	Prec↑	Rec↑	F-score↑	vs.**Open3D*↑**
Monodepth2 [19]	10.8	21.3	0.334	0.204	0.252	+0.018
DPT-ScanNet	9.4	13.3	0.412	0.361	0.384	+0.048
w sparse inputs	6.2	8.5	0.566	0.522	0.542	+0.065
Omnidata [10]	5.8	7.7	0.574	0.549	0.560	+0.069
Depth sensor	4.3	5.0	0.740	0.746	0.742	/

Table 6. **7Scenes reconstruction with varying depth priors**. The same DG-Recon model was compared utilizing different depths ranked by quality from low to high. None of the models was trained on 7Scenes or other Kinect data. Note that Omnidata outputs relative depth. Depth scale and offset was fit per frame with the DPT-ScanNet prediction to output metric depth. *F-score gain of DG-Recon comparing to Open3D TSDF integration.

While DG-Recon benefits from depth guidance, it might also be affected by the depth quality. Table 6 compares DG-Recon performance adopting different depth priors, varying from degraded depth models, off-the-shelf SOTA monocular depth to depth sensor readings. Even though lower accuracy depth models (Monodepth2 and vanila DPT-ScanNet) can negatively impact the performance, a SOTA depth model (Omnidata) is shown to improve DG-Recon's generalization capability to 7Scenes (F-score 0.560 vs. 0.542).

5.6. Limitations

Compared to the NeuralRecon baseline, DG-Recon introduces computational overhead because of the standalone depth network and the cross-attention-based fusion. The former might be optimized, as future improvements, by sharing the backbone between the depth estimation network and the image feature extractor of DG-Recon. The latter might be remedied by compromising slightly the reconstruction quality with the variance-based fusion.

Like other learning-based reconstruction methods, DG-Recon was trained using the ScanNet data captured with a single camera setup only. The reconstruction performance might drop as the target camera setup drifts significantly, *e.g*. with a larger field of view or heavier distortion. The sparse depth inputs already help mitigate the negative impact of camera differences and one could further fine-tune the depth model for the target camera setup without requiring any ground truth thanks to the recent progress of self-supervised depth estimation.

Another limitation of DG-Recon is the generalization capability to outdoor scenes. The distribution drift from indoor training data to outdoor test data might lead to accuracy degradation like other data-driven methods. An off-the-shelf monocular depth model could partially mitigate this shift thanks to DG-Recon's modular design, *i.e*., separated depth prior and TSDF prediction as shown by the example in supplementary material Section 5.

6. Conclusion

We present a real-time neural reconstruction method, DG-Recon, which improves image feature back projection and cross-view association with the guidance of depth priors. Together with the efficient variance- or cross-attention-based fusion modules, DG-Recon models can better express the geometric information and produce reconstructions with more details than the NeuralRecon baseline. Our method performs online reconstruction in real-time and achieves state-of-the-art reconstruction performance on the ScanNet dataset. With robust depth priors, DG-Recon generalizes better to the 7-Scenes and SUN3D datasets than the other state-of-the-art neural 3D reconstruction methods.

7. Acknowledgement

We would like to thank Gerhard Reitmayr, Abhijeet Bisain, Eduardo Esteves, Shang-Lun Lee, Yu Fu, and other colleagues for their support and valuable inputs to the project. We also highly appreciate the constructive feedback from reviewers on improving the manuscript.

References

[1] Aljaz Bozic, Pablo Palafox, Justus Thies, Angela Dai, and Matthias Nießner. Transformerfusion: Monocular rgb scene reconstruction using transformers. *NeurIPS*, 2021. 1, 2, 4, 5, 6, 8

[2] 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 *CVPR*, 2021. 3

[3] Zhao Chen, Vijay Badrinarayanan, Gilad Drozdov, and Andrew Rabinovich. Estimating depth from rgb and sparse sensing. In *ECCV*, 2018. 5

[4] Jaesung Choe, Sunghoon Im, Francois Rameau, Minjun Kang, and In So Kweon. Volumefusion: Deep depth fusion for 3d scene reconstruction. In *ICCV*, 2021. 2

[5] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *CVPR*, 2017. 2, 5, 7

[6] Angela Dai, Matthias Nießner, Michael Zollhöfer, Shahram Izadi, and Christian Theobalt. Bundlefusion: Real-time globally consistent 3d reconstruction using on-the-fly surface reintegration. *ACM TOG*, 2017. 1, 3

[7] Kangle Deng, Andrew Liu, Jun-Yan Zhu, and Deva Ramanan. Depth-supervised nerf: Fewer views and faster training for free. In *CVPR*, 2022. 3

[8] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 2

[9] Arda Duzceker, Silvano Galliani, Christoph Vogel, Pablo Speciale, Mihai Dusmanu, and Marc Pollefeys. Deepvideomvs: Multi-view stereo on video with recurrent spatio-temporal fusion. In *CVPR*, 2021. 2, 5

[10] Eftekhar et al. Omnidata. In *ICCV*, 2021. 9

[11] David Eigen and Rob Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *ICCV*, 2015. 2, 5

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

[13] Ziyue Feng, Leon Yang, Pengsheng Guo, and Bing Li. Cvrecon: Rethinking 3d geometric feature learning for neural reconstruction. 2023. 3

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

[15] Yasutaka Furukawa, Carlos Hernández, et al. Multi-view stereo: A tutorial. *Found. Trends Comput. Graph. Vis.*, 2015. 2

[16] Huiyu Gao, Wei Mao, and Miaomiao Liu. Visfusion. In *CVPR*, 2023. 3

[17] Ben Glocker, Shahram Izadi, Jamie Shotton, and Antonio Criminisi. Real-time rgb-d camera relocalization. In *ISMAR*, 2013. 2, 5, 6, 8

[18] Clément Godard, Oisin Mac Aodha, and Gabriel J Brostow. Unsupervised monocular depth estimation with left-right consistency. In *CVPR*, 2017. 5

[19] Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J Brostow. Digging into self-supervised monocular depth estimation. In *ICCV*, 2019. 2, 5, 9

[20] 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*, 2022. 3

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 5

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

[23] Sunghoon Im, Hae-Gon Jeon, Steve Lin, and In So Kweon. Dpsnet: End-to-end deep plane sweep stereo. In *ICLR*, 2019. 2

[24] Iro Laina, Christian Rupprecht, Vasileios Belagiannis, Federico Tombari, and Nassir Navab. Deeper depth prediction with fully convolutional residual networks. In *3DV*, 2016. 2, 5

[25] Fayao Liu, Chunhua Shen, and Guosheng Lin. Deep convolutional neural fields for depth estimation from a single image. In *CVPR*, 2015. 2

[26] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. *ACM SIGGRAPH*, 1987. 1, 5

[27] Fangchang Ma and Sertac Karaman. Sparse-to-dense: Depth prediction from sparse depth samples and a single image. In *ICRA*, 2018. 5

[28] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Commun. ACM*, 2021. 2

[29] Hans Moravec and Alberto Elfes. High resolution maps from wide angle sonar. In *ICRA*, 1985. 4

[30] Zak Murez, Tarrence Van As, James Bartolozzi, Ayan Sinha, Vijay Badrinarayanan, and Andrew Rabinovich. Atlas: End-to-end 3d scene reconstruction from posed images. In *ECCV*, 2020. 1, 2, 4, 5, 6, 7, 8

[31] Richard A Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J Davison, Pushmeet Kohi, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *ISMAR*, 2011. 1, 2

[32] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *ICCV*, 2021. 2, 5

[33] Alexander Rich, Noah Stier, Pradeep Sen, and Tobias Höllerer. 3dvnet: Multi-view depth prediction and volumetric refinement. In *3DV*, 2021. 2, 5, 6

[34] Barbara Roessle, Jonathan T Barron, Ben Mildenhall, Pratul P Srinivasan, and Matthias Nießner. Dense depth priors for neural radiance fields from sparse input views. In *CVPR*, 2022. 3

[35] Ashutosh Saxena, Sung Chung, and Andrew Ng. Learning depth from single monocular images. *NeurIPS*, 2005. 2

[36] Mohamed Sayed, John Gibson, Jamie Watson, Victor Prisacariu, Michael Firman, and Clément Godard. Simplerecon: 3d reconstruction without 3d convolutions. In *ECCV*, 2022. 2, 3, 5, 6, 8

[37] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *CVPR*, 2016. 3, 5

[38] Johannes L Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *ECCV*, 2016. 2

[39] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *ECCV*, 2016. 2

[40] 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. 2023. 3

[41] Noah Stier, Alexander Rich, Pradeep Sen, and Tobias Höllerer. Vortx: Volumetric 3d reconstruction with transformers for voxelwise view selection and fusion. In *3DV*, 2021. 1, 2, 3, 4, 5, 6, 7

[42] Jiaming Sun, Yiming Xie, Linghao Chen, Xiaowei Zhou, and Hujun Bao. Neuralrecon: Real-time coherent 3d reconstruction from monocular video. In *CVPR*, 2021. 1, 2, 3, 4, 5, 6, 7, 8

[43] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 2017. 4

[44] Kaixuan Wang and Shaojie Shen. Mvdepthnet: Real-time multiview depth estimation neural network. In *3DV*, 2018. 2

[45] 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. In *NeurIPS*, 2021. 3

[46] Silvan Weder, Johannes L. Schönberger, Marc Pollefeys, and Martin R. Oswald. Routedfusion: Learning real-time depth map fusion. In *CVPR*, 2020. 2

[47] Silvan Weder, Johannes L Schonberger, Marc Pollefeys, and Martin R Oswald. Neuralfusion: Online depth fusion in latent space. In *CVPR*, 2021. 2

[48] 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 *CVPR*, 2021. 3

[49] Cho-Ying Wu, Jialiang Wang, Michael Hall, Ulrich Neumann, and Shuochen Su. Toward practical monocular indoor depth estimation. In *CVPR*, 2022. 2, 5

[50] Jianxiong Xiao, Andrew Owens, and Antonio Torralba. Sun3d: A database of big spaces reconstructed using sfm and object labels. In *ICCV*, 2013. 2, 5, 6, 8

[51] Yan Xu, Xinge Zhu, Jianping Shi, Guofeng Zhang, Hujun Bao, and Hongsheng Li. Depth completion from sparse lidar data with depth-normal constraints. In *ICCV*, 2019. 5

[52] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. In *NeurIPS*, 2021. 3

[53] Andy Zeng, Shuran Song, Matthias Nießner, Matthew Fisher, Jianxiong Xiao, and Thomas Funkhouser. 3dmatch: Learning local geometric descriptors from rgb-d reconstructions. In *CVPR*, 2017. 5

[54] Xiaoshuai Zhang, Sai Bi, Kalyan Sunkavalli, Hao Su, and Zexiang Xu. Nerfusion: Fusing radiance fields for large-scale scene reconstruction. In *CVPR*, 2022. 3

[55] Chaoqiang Zhao, Youmin Zhang, Matteo Poggi, Fabio Tosi, Xianda Guo, Zheng Zhu, Guan Huang, Yang Tang, and Stefano Mattoccia. Monovit: Self-supervised monocular depth estimation with a vision transformer. *ArXiv*, 2022. 2

[56] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Open3D: A modern library for 3D data processing. *ArXiv*, 2018. 2, 6

[57] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G Lowe. Unsupervised learning of depth and ego-motion from video. In *CVPR*, 2017. 5

[58] Xingxing Zuo, Nan Yang, Nathaniel Merrill, Binbin Xu, and Stefan Leutenegger. Incremental dense reconstruction from monocular video with guided sparse feature fusion. *RA-L*, 2023. 3