Flora: Dual-Frequency LOss-Compensated ReAl-Time Monocular 3D Video Reconstruction

Likang Wang[1], Yue Gong[3], Qirui Wang[3], Kaixuan Zhou[4,5], Lei Chen[1,2]
[1]Department of Computer Science and Engineering, The Hong Kong University of Science and Technology
[2]Data Science and Analytics Thrust, The Hong Kong University of Science and Technology (Guangzhou)
[3]Distributed and Parallel Software Lab, Huawei Technologies
[4]Riemann Lab, Huawei Technologies
[5]Fundamental Software Innovation Lab, Huawei Technologies
lwangcg@connect.ust.hk, gongyue1@huawei.com, wangqirui1@huawei.com,
zhoukaixuan2@huawei.com, leichen@cse.ust.hk

Abstract

In this work, we propose a real-time monocular 3D video reconstruction approach named Flora for reconstructing delicate and complete 3D scenes from RGB video sequences in an end-to-end manner. Specifically, we introduce a novel method with two main contributions. Firstly, the proposed feature aggregation module retains both color and reliability in a dual-frequency form. Secondly, the loss compensation module solves missing structure by correcting losses for falsely pruned voxels. The dual-frequency feature aggregation module enhances reconstruction quality in both precision and recall, and the loss compensation module benefits the recall. Notably, both proposed contributions achieve great results with negligible inferencing overhead. Our state-of-the-art experimental results on real-world datasets demonstrate Flora's leading performance in both effectiveness and efficiency. The code is available at https://github.com/NoOneUST/Flora.

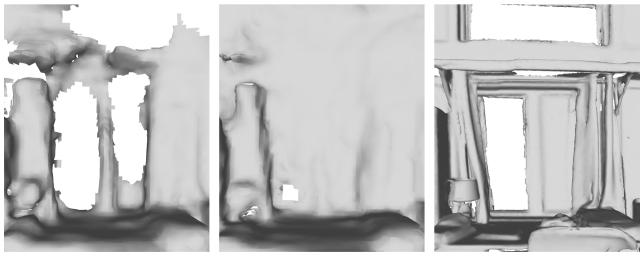

Figure 1: Defects of existing methods. Current SOTA (NeuralRecon by Sun et al.) loses structures, while our method leads to complete reconstructions. For example, the windows disappear in the current SOTA and even the ground truth from depth sensors. On the contrary, these windows are intact in our model.

1 Introduction

3D dense reconstruction has been one of the most fundamental ways to model the real world for the past decades and is becoming more and more crucial due to its fundamental place in robotics and virtual reality (VR). As real-life applications for 3D reconstruction develop, diverse technical requirements for 3D reconstruction emerge. For instance, the 3D reconstruction based on images or videos has become a mainstream solution for its usage of the most accessible, low-cost, and convenient sensor, the camera. Furthermore, the demand for real-time reconstruction was also becoming intense as many applications such as robotic navigation, virtual reality, and augmented reality(AR) need a real-time response. However, achieving real-time performance while retaining decent reconstruction quality has been challenging.

1.1 Taxonomy of Reconstruction Methods

Most RGB-based real-time 3D reconstruction methods are based on deep neural networks (Zeng et al. 2022, 2023; Zhang et al. 2023; Liang et al. 2022; Liu et al. 2022c) because of their remarkable success in various tasks, including natural language (Devlin et al. 2019; Brown et al. 2020), computer vision (Wang and Chen 2023b; Zhang et al. 2022; Li et al. 2022; Wang et al. 2023), and graph mining (Liu et al. 2022a,b, 2023d). Specifically, the existing approaches can generally be divided into depth-map-based and volume-based. The former genre (e.g., DeepVideoMVS by Duzceker et al.; Wang and Chen) first predicts depth maps from different views and then separately fuses depth predictions to generate the 3D representation (e.g., meshes and point clouds). In contrast, the latter (e.g., NeuralRecon by Sun et al.) first maps feature from different views to a common 3D space, then based on which a 3D representation is directly predicted in an end-to-end manner. Typically, the latter is superior in the synthesized quality (F-score) of the final 3D model (mesh) because it directly optimizes the 3D representation (e.g., signed distance function).

1.2 Taxonomy of Real-time Reconstruction

Real-time volumetric methods (e.g., NeuralRecon) usually split video frames into fragments. Afterward, multi-scale feature maps are extracted from each fragment to enable hierarchical reconstruction. At each scale, the model maps all the extracted 2D feature maps into a common 3D space and

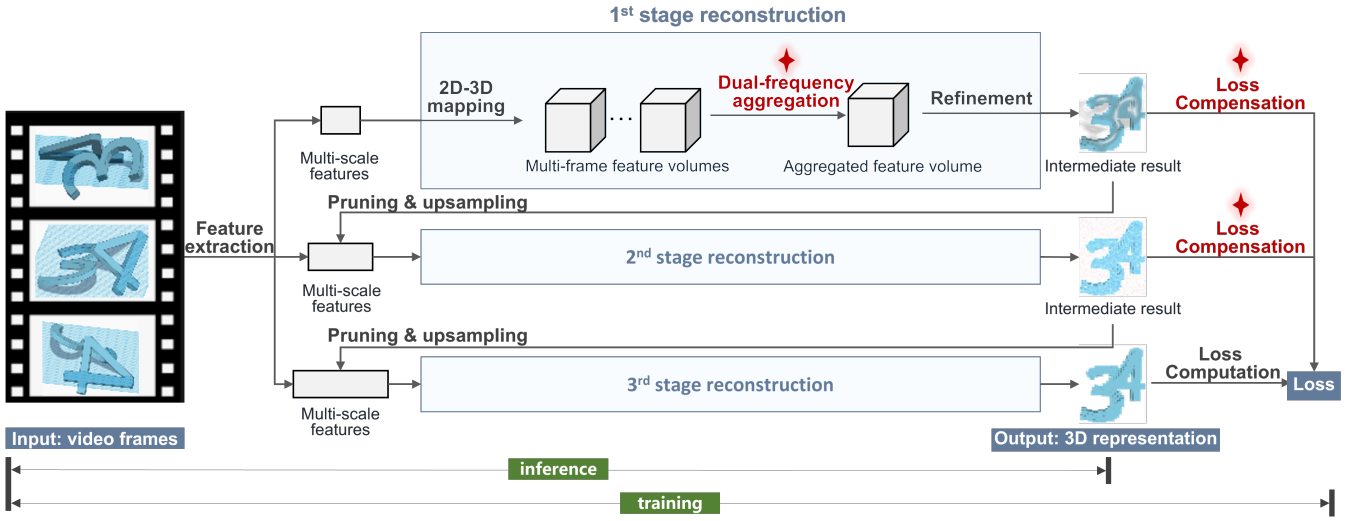

Figure 2: Model structure of Flora. Features at multiple resolutions are extracted from the input video sequences. In each scale, feature volumes from different frames are constructed and then fused by our dual-frequency aggregation module. After that, an implicit 3D representation is estimated from the sparsified integrated feature. The losses are calculated as the difference between the 3D prediction and the ground truth and then rectified by the loss compensation module.

then forms a feature volume for each frame. After that, all the feature volumes are aggregated into one, which is further refined to generate a 3D prediction. Then, the scale-wise loss is calculated as the difference between the prediction and ground truth. Next, the voxels are pruned according to the prediction confidence. Finally, the pruned voxels are upsampled, then sent to the higher-resolution scales.

1.3 Problems of Existing Methods

However, these methods usually contain holes and are defective in delicate structures. We argue that there are two main reasons listed below:

1. **Unreliable multi-frame aggregation:** Given preliminary feature volumes from different frames, utilization of the rich information influences the reconstruction quality significantly. However, aggregating information from multiple frames can be tricky and easily jeopardized by environmental flaws such as occlusions and adverse viewpoint changes.

   The existing methods aggregate features from multiple frames considering either color (e.g., Neuralrecon) or reliability (e.g., DeepVideoMVS). Specifically, DeepVideoMVS and IS-MVSNet (Wang et al. 2022) fuse warped views by computing their correlation and variance, respectively. The fused feature represents each depth's reliability because the correct depth leads to higher correlation and lower variance among warped views. However, these operations lose the original color information inside the RGB inputs. Since color is critical to semantic understanding and semantics is indispensable to reconstructing texture-less regions, these methods generally lead to curved or broken large surfaces.

   On the opposite, NeuralRecon and Atlas (Murez et al. 2020) aggregate views by averaging features from different viewpoints. Averaging helps to retain the original color information because the correct depth leads to similar colors among warped views, and the mean of similar colors is close to the input. However, it is hard to distinguish which color corresponds to the correct depth. Consequently, these methods result in less reliable depth predictions and less meticulous details in affluent-texture areas.

2. **Radical and incorrect voxel pruning:** Volume-based methods (Sun et al. 2021) generally adopt pruning to reduce computational overhead and utilize real-world scenes' sparse nature. Pruning is almost inevitable due to the hefty cost of computations on dense volumes; however, it leads to a new problem: the model tends to produce radical and incorrect voxel pruning.

   To specify, a coarse voxel's existence may be hard to distinguish because it covers a large area. Figure 3 shows that the model is puzzled on coarse voxels, thus assigning them confidence around the pruning threshold. If the confidence is lower than the threshold, then the voxel is pruned, thus having no loss contribution in finer stages; otherwise, the voxel is retained and produces significant losses in finer stages. In other words, although reducing the confidence of a coarse voxel, which should be reserved, slightly increases the low-resolution stage loss; however, the higher-resolution stage loss gets dramatically reduced. Since the model is optimized toward a smaller total loss accumulated at all stages, it tends to falsely and radically prune voxels, resulting in incomplete reconstructions.

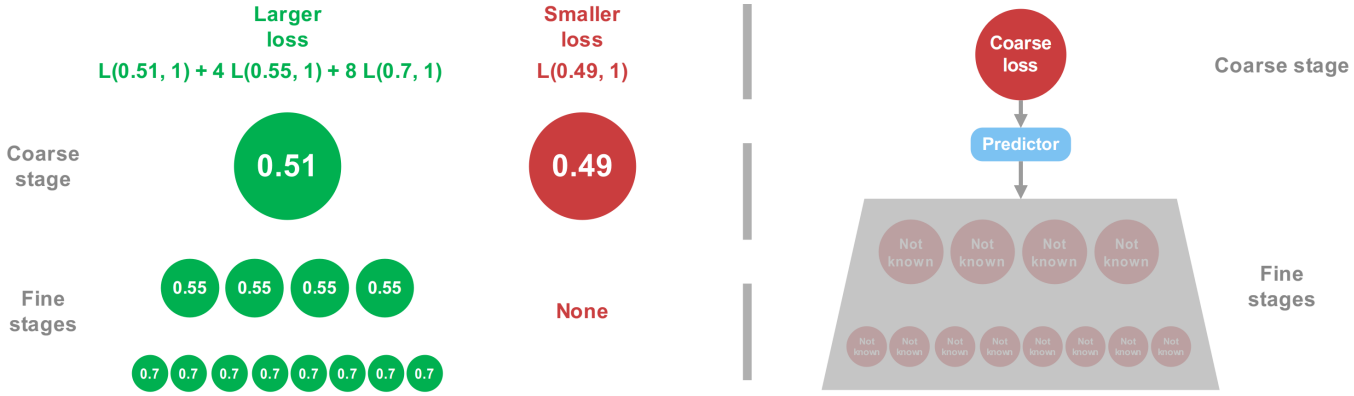

Figure 3: Loss compensation. Left (a): Radical and incorrect voxel pruning problem. Give a puzzling voxel; if the confidence prediction (0.51) is higher than the pruning threshold (0.5), it will be retained and up-sampled to generate descendant voxels. If the estimation is smaller (0.49), it is pruned and produces no offspring in the finer stages. Since the total loss considers voxels in all phases, the loss will reduce after pruning positive voxels. Right (b): Our loss compensation module predicts finer stage losses based on the coarse stage loss for falsely pruned voxels.

1.4 Our Contributions

This paper proposes a real-time monocular 3D video reconstruction method called Flora, the abbreviation of dual-Frequency LOss-compensated ReAl-time 3D reconstruction, which addresses the two challenges mentioned above by two novel contributions below:

1. **Dual-frequency aggregation:** Considering the first challenge, we propose considering both color and reliability because color is essential to smooth surfaces, and reliability is crucial in rich-texture regions. Instead of processing these two modalities separately, we allow them to collaborate. To specify, we use multi-layer CNNs to allow semantic propagation from more reliable to less reliable areas. This way, both non-textured regions (walls, grounds) and delicate structures are better recovered.

   We categorize averaging into low-pass filters and variance / correlation into high-pass filters because low-pass filters (e.g., Gaussian) retain the commons and the main structures of inputs. In contrast, high-pass filters (e.g., Sobel 1970, Canny 1986) look for discrepancies and details. We propose to extract color by low-pass filters and reliability by high-pass filters, thus calling this module dual-frequency aggregation.

   Although there are many ways (e.g., Fourier, wavelets transformation) to extract different frequency components, most require long signal sequences and high computation demand. However, the number of views is always minimal (smaller than ten), and the system latency is crucial to real-time reconstruction. We show that combining the most naive and lightweight filters (averaging and the L^1 difference) already leads to significant effectiveness benefits while retaining state-of-the-art efficiency.

   Although averaging and the L^1 difference are off-the-self operators, revealing their complemental physical nature is contributive. Moreover, we provide a highly efficient way to decompose multi-view data utilizing geometric priors. In addition, we suggest a possible direction to look for better operators by introducing the concept of low- and high-pass filters.

2. **Loss compensation:** The second challenge significantly impacts the model training. The difficulty is to efficiently estimate the losses of all the descendants of each falsely pruned voxel. However, we cannot directly measure the loss of a voxel whose ancestor has been pruned. Thus, we propose to estimate the finer stage losses based on their ancestors (i.e., the coarse stage loss for each falsely pruned voxel) using a simple function. We demonstrate that the mapping function can be accurately regressed from the data. In this way, falsely pruning voxels becomes unfavored to the model because it leads to a higher coarse stage loss and identical finer stage losses. This strategy promotes completeness (recall) and is cost-free in testing.

1.5 Summary

To summarize, we have two contributions:

- We efficiently extract and aggregate two modalities: color and reliability, from the video sequence. Our method leads to significant improvements in both precision and recall while retaining state-of-the-art efficiency.
- We significantly ease the incorrect voxel pruning problem by estimating the loss of the descendants of each falsely pruned voxel. Our method greatly improves recall while retaining state-of-the-art efficiency and precision.

Our method performs better on both precision and recall while retaining efficiency, equipped with both the dual frequency aggregation module and the loss compensation module. Our experiments on real-world datasets corroborate our strategies' effectiveness. On ScanNet, we achieve an F-score of 58.4%, 2.2% higher than the current SOTA (Neuralrecon), while running in real-time at an FPS of 30 on one single RTX 2080 Ti GPU.

2 Related Work

This section first discusses multi-view stereo (MVS) because most monocular reconstruction methods are developed from MVS. Then, we present state-of-the-art monocular video reconstruction methods. Finally, we discuss neural radiance fields because they also construct implicit representations.

2.1 Multi-view Stereo (MVS)

Most MVS methods focus on offline reconstruction from multiple views. The first learning-based MVS approach is SurfaceNet (2017), a divide-and-conquer framework based on 3D convolutional networks. MVSNet (2018) eases the spatial demand by first predicting depth maps in a differentiable way and then fusing them into 3D point clouds. MVSNet aggregates view images using their variance, and many follow-up works (Wang et al. 2022; Gu et al. 2020; Zhang et al. 2020) adopt the same or similar solutions (e.g., correlation). This strategy works well as a reliability extraction backbone in our model, although our reliability branch adopts the L^1 difference in default for simplicity and higher efficiency.

2.2 Monocular Video Reconstruction

Instead of images captured from multiple views (Liu et al. 2023a,b), monocular video reconstruction takes temporally continuous video frames (Liu et al. 2023c) as input. There are mainly two technical streams: depth-map-based and volume-based.

Volumetric: As a volume-based method, Atlas (Murez et al. 2020) fuses features from the entire video into a volume, which is further decoded to recover the whole scenario. VoRTX (Stier et al. 2021) models temporal relationship with transformer. TransformerFusion (Prakash, Chitta, and Geiger 2021) uses transformers to select frames. NeuralRecon (Sun et al. 2021) converts Atlas into a real-time approach by incrementally processing video frames within a sliding window and sparsification.

Depth-based: As for the depth-map-based method, MVDepthNet (Wang and Shen 2018) constructs 2D cost volumes and then processes them with 2D convolutions. CNMNet (Long et al. 2020) considers planar information to make depth smooth. DeepvideoMVS (Duzceker et al. 2021) temporally propagates information by fusing the warped former predictions with current features. ESTDepth (Long et al. 2021) models spatial and temporal correlation with transformers. 3DVNet(Rich et al. 2021) utilizes 3D CNNs to process depth maps jointly.

Real-time: Among all the mentioned methods, only NeuralRecon and DeepVideoMVS infer in real-time (more than 30 fps) on typical GPUs (e.g., RTX 2080 Ti) while archiving relatively high effectiveness. Considering NeuralRecon's advantages on the final 3D model's quality, we inherit the coarse-to-fine pipeline of NeuralRecon. However, we are significantly different in integrating the back-projected view features and calculating the loss function.

2.3 Neural Radiance Fields

Since the proposal of NeRF (Mildenhall et al. 2020), neural radiance fields have gained tremendous success in recent two years. Like the volumetric reconstruction methods, neural radiance field methods also learn implicit representations from data. The significant difference is that neural radiance field methods (Saito et al. 2019) use coordinates to query the network for results, while volumetric reconstruction methods predict truncated signed distance function (TSDF) representations.

3 Methodology

Our macro pipeline is inherited from NeuralRecon; thus, we first describe the latter in Section 3.1. Afterward, Sections 3.2 and 3.3 propose our main contributions: novel loss rectification and feature aggregation strategies.

Inputs:

1. Temporally continuous images $\{\mathbf{I}_t\}$ extracted from a video sequence, where $t \in \mathbb{N}$ denotes the current timestamp.
2. The corresponding camera poses $\{\xi_t\}$ estimated by an off-the-shelf simultaneous localization and mapping (SLAM) solution.

Goal: Recover the 3D representation of the scenario described by the input video sequence.

3.1 Macro Pipeline Inherited from NeuralRecon

Following NeuralRecon, our model reconstructs in a *coarse-to-fine* manner. We temporally divide video frames into non-overlapped fragments and process them one by one. All images within the current fragment are sent to a feature pyramid network (FPN) (Lin et al. 2017) for multi-scale representation extraction. In each scale, the 2D features from each view are first back-projected into 3D by the *2D-to-3D projection* module, then fused into a feature volume by the *feature aggregation* module. After that, the *GRU fusion* module is applied to keep consistent along the temporal and spatial dimensions. Next, we sparsify the voxels with the *pruning* module. Finally, the error is back-propagated based on the *loss function* module.

Coarse-to-fine: We form a voxel volume in the coarsest stage $s = 0$ by sampling from a uniform 3D grid, determined by the field of view of all the frames within the input fragment. In the finer stages $s \geq 1$, we up-sample eight new voxels around each voxel that survived after pruning. The number is eight because we have three dimensions total, and each dimension is up-sampled by two times. For each up-sampled voxel at stage $s+1$, its updated feature F'_{s+1} is the concatenation of the back-projected feature F_{s+1}, the signed distance function (SDF), and the occupancy. We maintain a global SDF volume and replace the corresponding areas using the sparsified SDF and occupancy predictions at the finest stage $s = 2$.

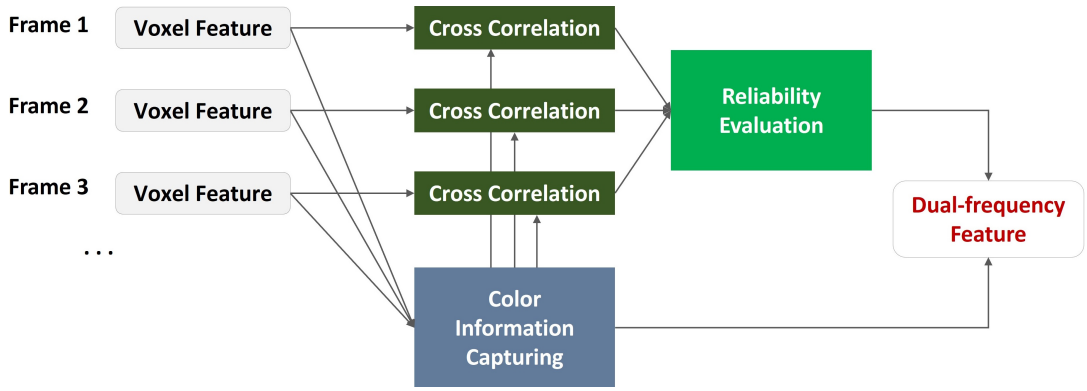

Figure 4: Our dual-frequency feature aggregation module. We first integrate the color information by averaging the neighboring frames and then extract the reliability information by differencing. Finally, the color and reliability information get fused by concatenation.

2D-to-3D projection: Given a voxel in the 3D space, if it is visible in one view, its projection is the intersection of this view's image plane and the line determined by the voxel and the camera's optical center. Since we have no idea in which views the voxel is visible, we collect pixel features from all the views. After operating on all the sampled voxels, we form n_v feature volumes, where n_v is the number of views.

*** Feature aggregation:** Its input is the output of the 2D-to-3D module, i.e., a set of feature volumes projected from different views; its output is the input of the GRU fusion module, i.e., one single aggregated feature volume. NeuralRecon calculates the mean of all views as the aggregated feature of each voxel. We integrate views by considering both color and reliability and illustrate the detailed approach in Section 3.2.

GRU fusion: The aggregated feature is first sent to a sparse 3D UNet (Çiçek et al. 2016) for spatial refinement. After this, we consider temporal consistency by a convolutional GRU module (Shi et al. 2015). In detail, we have a sparse volume to store the whole scenario's 3D representation, from which we extract the voxels within the current feature volume's field of vision. Thanks to sparsification, the volume does not need much space to store. Then, the previous predictions and current features are fused by a ConvGRU. The updated hidden state replaces the corresponding area within the global volume. The model estimates two volumes at each stage: occupancy and signed distance function (SDF). Occupancy $\in [0, 1]$ denotes whether the voxel exists, and SDF represents the voxel's distance to the nearest surface.

Pruning: Increasing the number of voxels to the cube, unavoidable for spatial resolution promotion, significantly increases the computation cost. Therefore, we sparsify the feature volumes and prune the seemingly unreliable voxels at the end of each stage after loss calculation. If a voxel's occupancy is estimated to be less than a threshold, it gets pruned. Otherwise, we retain it. The threshold is set as 0.5 to balance positive and negative samples.

*** Loss function:** Its input is the output of the MLP prediction heads, i.e., a confidence volume representing the existence of voxels and an SDF volume representing each voxel's distance to the nearest surface. We calculate the confidence loss by a binary cross-entropy function and the distance loss by a L^1 function. After obtaining losses at all stages, we sum them up, then use the summed loss to calculate the gradients. NeuralRecon calculates loss only for the voxels that survived after pruning. We efficiently measure all voxels' losses using our novel loss compensation strategy elaborated in Section 3.3.

3.2 * Dual-frequency Feature Aggregation

Given a voxel in the feature volume, its representation $\{V_i\}$ in frame $i \in [1, n_v]$, n_v denoting the number of frames, is provided by the 2D-to-3D projection. Since color is important to plane smoothness and reliability is crucial to distinguishing surface and void, we aim to fuse all frame features $\{V_i\}$ into one single feature F containing both color and reliability.

As discussed in Section 1, color and reliability can be extracted using low- and high-pass filters, respectively.

Low-pass: We use a low-pass filter to aggregate $\{V_i\}$ for retaining the color. For computational efficiency, we adopt averaging as the low-pass filter. Then, the aggregated color feature $C = \frac{1}{n_v} \sum_{i=1}^{n_v} V_i$. Since $\{V_i\}$ coincide at the correct depth, averaging them retains the original color.

High-pass: We use a high-pass filter to aggregate $\{V_i\}$ for reliability measures. For efficiency, we adopt the L^1 difference as the high-pass filter. Then, the aggregated reliability feature $R = \frac{1}{n_v} \sum_{i=1}^{n_v} |V_i - C|$. Suppose the voxel is on a surface and is visible; then the image colors should be the same because they are the correct projections of the voxel. In consequence, the difference between them is small. Otherwise, the difference is large. Thus, this operation estimates reliability.

Fuse: Finally, we concatenate the color C and the reliability R along the channel dimension to obtain the aggregated

Method	Abs Rel↓	Abs Diff↓	RMSE↓	Prec↑	Recall↑	**F-score↑**	**Time (ms)↓**
MVDepthNet (Wang and Shen)	0.10	0.19	0.29	0.208	0.831	0.329	48
GPMVS (Hou, Kannala, and Solin)	0.13	0.24	0.47	0.188	**0.871**	0.304	51
DPSNet (Im et al.)	0.09	0.16	0.23	0.223	0.793	0.344	322
COLMAP (Schönberger et al.)	0.14	0.26	0.50	0.505	0.634	0.558	2076
DeepVideoMVS (Duzceker et al.)	0.07	0.12	0.21	0.439	0.477	0.450	32
NeuralRecon (Sun et al.)	0.07	0.11	0.20	0.684	0.479	0.562	**30**
Ours	**0.06**	**0.09**	**0.19**	**0.701**	0.504	**0.584**	**30**

Table 1: Results on ScanNet following the experimental settings of NeuralRecon. The most important metrics are the F-score and the inference time. The results of MVDepthNet, GPMVS, DPSNet, COLMAP, and NeuralRecon are inherited from the paper of NeuralRecon. Our method ranks top in efficiency while significantly outperforming existing real-time methods (e.g., NeuralRecon, DeepVideoMVS) in effectiveness. Our real-time method is even better than some offline methods (e.g., COLMAP and DPSNet). The best method in each metric is marked in bold.

feature $F = C \| R$. Notably, the input frame features $\{V_i\}$'s channel number is reduced to half to avoid doubling the aggregated feature F's size.

In this way, color and reliability can be considered simultaneously. The model can utilize color to recover planes and reliability to tackle rich-texture regions. Further, we argue that color and reliability remedy defects for each other because the following multi-layer CNNs (3D UNet by Çiçek et al.) have the potential to propagate color features from reliable to unreliable regions using its increasing receptive field.

3.3 * Loss Compensation

As illustrated in Figure 3, sparse pruning strategies lead to a new problem: a helpful voxel pruned at coarse stages cannot be recovered at finer stages. Notably, spatial resolution at coarse stages is relatively low; thus, it may not be informative and distinguishable enough to identify the existence of voxels sufficiently. To this end, the model's completeness is restricted and cannot work well on delicate structures.

Macro idea: To avoid this problem, we must obtain the loss for all voxels at all stages. However, directly calculating all the losses requires dense prediction, which is impossible due to the high computation cost. To retain the high efficiency of sparsification and the high effectiveness of dense prediction, we propose to estimate a falsely pruned voxel's total loss based on its coarse stage loss.

Mapping function: Given an actually positive voxel p_m and suppose it has loss l_m at stage m and is falsely pruned at stage $m < q$, q denoting the total number of stages. Then, our goal is to estimate the sum of its finer stage losses $L_f = \sum_{j=m+1}^{q} \ell_j$, ℓ_j denoting the sum of losses of all voxels p_j at stage j up-sampled from voxel p_m. We assume the mapping function from l_m to L_f is g, i.e., $L_f = g(l_m)$. Then, we analyze the property of g.

Coefficient estimation: Since voxels $\in \{p_{m+1}\}$ at stage $m+1$ are uniformly sampled around p_m, each voxel $\in \{p_{m+1}\}$ has an equal chance to be more distant or closer to the surface than p_m. This implies that it is reasonable to assume the average loss of voxels $\in \{p_{m+1}\}$ close to l_m. Then, we can use a linear function to model g.

The next task is determining the linear coefficient e_g of g. The compensation for falsely pruned voxels is insufficient when e_g is too small. In consequence, the model tends to wrong good pixels. At the same time, when e_g is too large, the model prefers preserving bad pixels to avoid accidentally injuring the good. In other words, both too small and too large are bad. In practice, we can efficiently use ternary search to find a good enough e_g. We use the F-score on the validation set to distinguish each e_g's quality. Our strategy does not impact testing efficiency because it is only related to loss calculation. Besides, it does not affect the detailed pruning strategy.

4 Experiments

We first illustrate the experimental settings in Section 4.1 and then show the results in Section 4.2.

4.1 Experimental Settings

Datasets: We test our model on the most popular indoor video reconstruction dataset: ScanNet (Dai et al. 2017). ScanNet is a large-scale dataset providing RGB video sequences and camera poses on more than 1600 indoor scenarios. We split ScanNet following its standard settings.

Model implementation: The model is constructed using PyTorch (2019), the sparse operations are implemented with TorchSparse (2022), and the 2D feature extraction module is a pre-trained MNasNet (2019) model.

Hyper-parameter: We adopt MADGRAD (Defazio and Jelassi 2022) as the optimizer, and the learning rate is set to 10^{-3}. Each feature volume is aggregated from nine views and is a cubic of sidelength [24, 48, 96] at three stages. Our hierarchical framework contains three layers in total to balance efficiency and effectiveness. The finest voxel size is 4cm, and the TSDF truncation distance is 12cm.

Baselines: We compare our method to both online and offline 3D reconstruction methods to demonstrate our method's strength in both effectiveness and efficiency. The chosen online algorithms cover both depth-map-based methods, e.g., DeepVideoMVS (Duzceker et al. 2021) (depth map-based SOTA), MVDepthNet (Wang and Shen 2018)

Method	Abs Rel↓	Abs Diff↓	RMSE↓	Prec↑	Recall↑	F-score↑
Baseline (Re-implementation of NeuralRecon)	0.07	0.10	0.20	0.688	0.472	0.558
Dual Frequency	**0.06**	**0.09**	**0.19**	**0.705**	0.488	0.574
Dual Frequency + Loss Compensation	**0.06**	**0.09**	**0.19**	0.701	**0.504**	**0.584**

Table 2: Impacts of each component. The reported results are measured on the ScanNet dataset. The baseline metrics correspond to the official weights of NeuralRecon.

and GPMVS (Hou, Kannala, and Solin 2019), and volume-based SOTA NeuralRecon (Sun et al. 2021). The selected offline methods include COLMAP (Schönberger et al. 2016) (one of the most popular) and DPSNet (Im et al. 2019).

Evaluation: We evaluate the 3D meshes following ATLAS (Murez et al. 2020) and measure the rendered depth maps using the 2D metrics introduced by Eigen et al. (2014). All 3D meshes are of a single layer and are obtained from the signed distance functions (SDF) by the marching cube algorithm (Lorensen and Cline 1987). The depth maps of depth-based algorithms are integrated into SDFs using the standard TSDF fusion (Zeng et al. 2017). The depth maps of volumetric algorithms are rendered by projection. Since our goal is to reconstruct the 3D structure and both precision and recall are valuable, we treat the F-score of the reconstructed mesh as the most critical metric.

4.2 Results

ScanNet: Table 1 shows that Flora is significantly better than existing real-time SOTAs in most metrics. We perform better or equally on all metrics than NeuralRecon, the most relevant volumetric SOTA method. Compared to DeepVideoMVS, the depth map-based SOTA, we also have significant superiority in most metrics, especially F-score and precision-related metrics. Our method is even superior to offline methods (COLMAP and DPSNet) in effectiveness. Moreover, the visualizations in Figure 3 further verify our advantages.

Generalize to unseen datasets: Our method not only shows advantages in public datasets, but also generalizes well to new scenarios. We train our model on ScanNet and then directly apply the trained weights to an unseen outdoor video sequence (Sun et al. 2021). As shown in Figure 5, our model correctly recovers the small garden.

5 Ablation Study

Impact of each component: As shown in Table 2, our dual-frequency feature aggregation benefits both precision and recall. In addition, our loss compensation module benefits recall.

Limitations: Like most 3D reconstruction methods, our model requires a large amount of training data. However, high-quality 3D data with known camera poses is expensive to acquire.

6 Conclusion

This paper proposes a real-time monocular 3D reconstruction method that is strong in both efficiency and effectiveness. Our contributions can be summarized into two points:

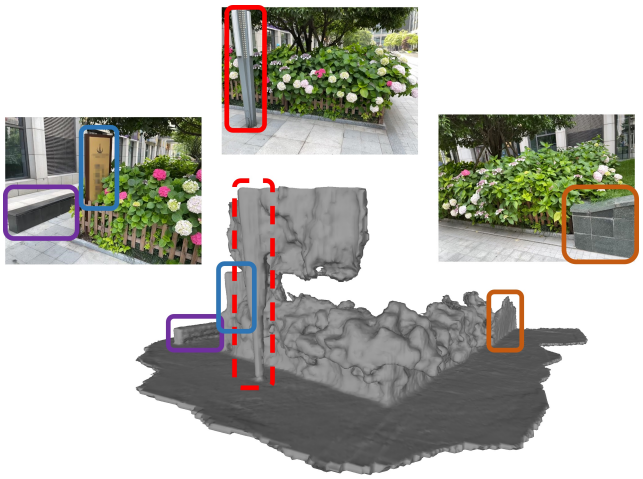

Figure 5: Our method generalizes well in the real world. The model is trained on ScanNet and tested on a new scenario captured by smartphones (Sun et al. 2021).

1. We propose a novel and efficient feature aggregation module considering color and reliability simultaneously.
2. We propose a novel loss compensation module avoiding false pruning.

These two strategies significantly and costlessly benefit the reconstruction quality in both precision and recall. Experiments on real-world data demonstrate our method's state-of-the-art performance and generalization.

Acknowledgments

This work is partially supported by the National Science Foundation of China (NSFC) under Grant No. U22B2060, the Hong Kong RGC GRF Project 16209519, CRF Project C6030-18G, C2004-21GF, AOE Project AoE/E-603/18, RIF Project R6020-19, Theme-based project TRS T41-603/20R, China NSFC No. 61729201, Guangdong Basic and Applied Basic Research Foundation 2019B151530001, Hong Kong ITC ITF grants MHX/078/21 and PRP/004/22FX, Microsoft Research Asia Collaborative Research Grant, HKUST-Webank joint research lab grant and HKUST Global Strategic Partnership Fund (2021 SJTU-HKUST). In addition, we gratefully acknowledge the generous support from https://www.mindspore.cn.

References

Brown, T. B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; Agarwal, S.; Herbert-Voss, A.; Krueger, G.; Henighan, T.; Child, R.; Ramesh, A.; Ziegler, D. M.; Wu, J.; Winter, C.; Hesse, C.; Chen, M.; Sigler, E.; Litwin, M.; Gray, S.; Chess, B.; Clark, J.; Berner, C.; McCandlish, S.; Radford, A.; Sutskever, I.; and Amodei, D. 2020. Language Models are Few-Shot Learners. In Larochelle, H.; Ranzato, M.; Hadsell, R.; Balcan, M.; and Lin, H., eds., *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Canny, J. 1986. A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, (6): 679–698.

Çiçek, Ö.; Abdulkadir, A.; Lienkamp, S. S.; Brox, T.; and Ronneberger, O. 2016. 3D U-Net: learning dense volumetric segmentation from sparse annotation. In *International conference on medical image computing and computer-assisted intervention*, 424–432. Springer.

Dai, A.; Chang, A. X.; Savva, M.; Halber, M.; Funkhouser, T.; and Nießner, M. 2017. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5828–5839.

Defazio, A.; and Jelassi, S. 2022. Adaptivity without compromise: a momentumized, adaptive, dual averaged gradient method for stochastic optimization. *Journal of Machine Learning Research*, 23: 1–34.

Devlin, J.; Chang, M.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Burstein, J.; Doran, C.; and Solorio, T., eds., *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, 4171–4186. Association for Computational Linguistics.

Duzceker, A.; Galliani, S.; Vogel, C.; Speciale, P.; Dusmanu, M.; and Pollefeys, M. 2021. DeepVideoMVS: Multi-view stereo on video with recurrent spatio-temporal fusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15324–15333.

Eigen, D.; Puhrsch, C.; and Fergus, R. 2014. Depth map prediction from a single image using a multi-scale deep network. *Advances in neural information processing systems*, 27.

Gu, X.; Fan, Z.; Zhu, S.; Dai, Z.; Tan, F.; and Tan, P. 2020. Cascade cost volume for high-resolution multi-view stereo and stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2495–2504.

Hou, Y.; Kannala, J.; and Solin, A. 2019. Multi-view stereo by temporal nonparametric fusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2651–2660.

Im, S.; Jeon, H.; Lin, S.; and Kweon, I. S. 2019. DPSNet: End-to-end Deep Plane Sweep Stereo. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Ji, M.; Gall, J.; Zheng, H.; Liu, Y.; and Fang, L. 2017. Surfacenet: An end-to-end 3d neural network for multiview stereopsis. In *Proceedings of the IEEE International Conference on Computer Vision*, 2307–2315.

Li, H.; Zhang, L.; Zhang, D.; Fu, L.; Yang, P.; and Zhang, J. 2022. TransVLAD: Focusing on Locally Aggregated Descriptors for Few-Shot Learning. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XX*, 524–540. Springer.

Liang, K.; Meng, L.; Liu, M.; Liu, Y.; Tu, W.; Wang, S.; Zhou, S.; Liu, X.; and Sun, F. 2022. Reasoning over Different Types of Knowledge Graphs: Static, Temporal and Multi-Modal. *arXiv preprint arXiv:2212.05767*.

Lin, T.-Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; and Belongie, S. 2017. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2117–2125.

Liu, C.; Wen, J.; Luo, X.; Huang, C.; Wu, Z.; and Xu, Y. 2023a. DICNet: Deep Instance-Level Contrastive Network for Double Incomplete Multi-View Multi-Label Classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Liu, C.; Wen, J.; Luo, X.; and Xu, Y. 2023b. Incomplete Multi-View Multi-Label Learning via Label-Guided Masked View- and Category-Aware Transformers. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Liu, L.; An, J.; Liu, J.; Yuan, S.; Chen, X.; Zhou, W.; Li, H.; Wang, Y.; and Tian, Q. 2023c. Low-Light Video Enhancement with Synthetic Event Guidance. *Proceedings of the AAAI Conference on Artificial Intelligence*.

Liu, Y.; Tu, W.; Zhou, S.; Liu, X.; Song, L.; Yang, X.; and Zhu, E. 2022a. Deep Graph Clustering via Dual Correlation Reduction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 7603–7611.

Liu, Y.; Xia, J.; Zhou, S.; Wang, S.; Guo, X.; Yang, X.; Liang, K.; Tu, W.; Li, Z. S.; and Liu, X. 2022b. A Survey of Deep Graph Clustering: Taxonomy, Challenge, and Application. *arXiv preprint arXiv:2211.12875*.

Liu, Y.; Yang, X.; Zhou, S.; and Liu, X. 2022c. Simple contrastive graph clustering. *arXiv preprint arXiv:2205.07865*.

Liu, Y.; Yang, X.; Zhou, S.; Liu, X.; Wang, Z.; Liang, K.; Tu, W.; Li, L.; Duan, J.; and Chen, C. 2023d. Hard Sample Aware Network for Contrastive Deep Graph Clustering. In *Proc. of AAAI*.

Long, X.; Liu, L.; Li, W.; Theobalt, C.; and Wang, W. 2021. Multi-view depth estimation using epipolar spatio-temporal networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8258–8267.

Long, X.; Liu, L.; Theobalt, C.; and Wang, W. 2020. Occlusion-aware depth estimation with adaptive normal constraints. In *European Conference on Computer Vision*, 640–657. Springer.

Lorensen, W. E.; and Cline, H. E. 1987. Marching cubes: A high resolution 3D surface construction algorithm. *ACM siggraph computer graphics*, 21(4): 163–169.

Mildenhall, B.; Srinivasan, P. P.; Tancik, M.; Barron, J. T.; Ramamoorthi, R.; and Ng, R. 2020. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European conference on computer vision*, 405–421. Springer.

Murez, Z.; As, T. v.; Bartolozzi, J.; Sinha, A.; Badrinarayanan, V.; and Rabinovich, A. 2020. Atlas: End-to-end 3d scene reconstruction from posed images. In *European Conference on Computer Vision*, 414–431. Springer.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

Prakash, A.; Chitta, K.; and Geiger, A. 2021. Multi-modal fusion transformer for end-to-end autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7077–7087.

Rich, A.; Stier, N.; Sen, P.; and Höllerer, T. 2021. 3DVNet: Multi-View Depth Prediction and Volumetric Refinement. In *2021 International Conference on 3D Vision (3DV)*, 700–709. IEEE.

Saito, S.; Huang, Z.; Natsume, R.; Morishima, S.; Kanazawa, A.; and Li, H. 2019. Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2304–2314.

Schönberger, J. L.; Zheng, E.; Frahm, J.; and Pollefeys, M. 2016. Pixelwise View Selection for Unstructured Multi-View Stereo. In Leibe, B.; Matas, J.; Sebe, N.; and Welling, M., eds., *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III*, volume 9907 of *Lecture Notes in Computer Science*, 501–518. Springer.

Shi, X.; Chen, Z.; Wang, H.; Yeung, D.-Y.; Wong, W.-K.; and Woo, W.-c. 2015. Convolutional LSTM network: A machine learning approach for precipitation nowcasting. *Advances in neural information processing systems*, 28.

Sobel, I. E. 1970. *Camera models and machine perception*. Stanford University.

Stier, N.; Rich, A.; Sen, P.; and Höllerer, T. 2021. VoRTX: Volumetric 3D reconstruction with transformers for voxel-wise view selection and fusion. In *2021 International Conference on 3D Vision (3DV)*, 320–330. IEEE.

Sun, J.; Xie, Y.; Chen, L.; Zhou, X.; and Bao, H. 2021. NeuralRecon: Real-time coherent 3D reconstruction from monocular video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15598–15607.

Tan, M.; Chen, B.; Pang, R.; Vasudevan, V.; Sandler, M.; Howard, A.; and Le, Q. V. 2019. Mnasnet: Platform-aware neural architecture search for mobile. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2820–2828.

Tang, H.; Liu, Z.; Li, X.; Lin, Y.; and Han, S. 2022. TorchSparse: Efficient Point Cloud Inference Engine. In *Conference on Machine Learning and Systems (MLSys)*.

Wang, K.; and Shen, S. 2018. Mvdepthnet: Real-time multiview depth estimation neural network. In *2018 International conference on 3d vision (3DV)*, 248–257. IEEE.

Wang, L.; and Chen, L. 2023a. Dionysus: Recovering Scene Structures by Dividing into Semantic Pieces. *ResearchGate*.

Wang, L.; and Chen, L. 2023b. FTSO: Effective NAS via First Topology Second Operator. *Preprints*.

Wang, L.; Gong, Y.; Ma, X.; Wang, Q.; Zhou, K.; and Chen, L. 2022. IS-MVSNet:Importance Sampling-Based MVS-Net. In Avidan, S.; Brostow, G.; Cissé, M.; Farinella, G. M.; and Hassner, T., eds., *Computer Vision – ECCV 2022*, 668–683. Cham: Springer Nature Switzerland. ISBN 978-3-031-19824-3.

Wang, S.; Yan, Z.; Zhang, D.; Wei, H.; Li, Z.; and Li, R. 2023. Prototype Knowledge Distillation for Medical Segmentation with Missing Modality. *arXiv preprint arXiv:2303.09830*.

Yao, Y.; Luo, Z.; Li, S.; Fang, T.; and Quan, L. 2018. MVS-Net: Depth Inference for Unstructured Multi-view Stereo. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

Zeng, A.; Song, S.; Nießner, M.; Fisher, M.; Xiao, J.; and Funkhouser, T. 2017. 3DMatch: Learning Local Geometric Descriptors from RGB-D Reconstructions. In *CVPR*.

Zeng, D.; Liu, W.; Chen, W.; Zhou, L.; Zhang, M.; and Qu, H. 2023. Substructure Aware Graph Neural Networks. In *Proc. of AAAI*.

Zeng, D.; Zhou, L.; Liu, W.; Qu, H.; and Chen, W. 2022. A Simple Graph Neural Network via Layer Sniffer. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 5687–5691. IEEE.

Zhang, D.; Chen, M.; Li, C.; Huang, L.; and Zhang, J. 2023. Aggregation of Disentanglement: Reconsidering Domain Variations in Domain Generalization. *arXiv preprint arXiv:2302.02350*.

Zhang, D.; Li, C.; Li, H.; Huang, W.; Huang, L.; and Zhang, J. 2022. Rethinking Alignment and Uniformity in Unsupervised Image Semantic Segmentation. *arXiv preprint arXiv:2211.14513*.

Zhang, J.; Yao, Y.; Li, S.; Luo, Z.; and Fang, T. 2020. Visibility-aware Multi-view Stereo Network. In *31st British Machine Vision Conference 2020, BMVC 2020, Virtual Event, UK, September 7-10, 2020*. BMVA Press.