Appendix (No.23-TIE-1236)

I. Related Work

A. Traditional Multi-View Stereo

In terms of output representation, traditional MVS methods can be partitioned into three categories: point cloud-based [1], [2], volumetric [3], [4], and depth map-based methods [5]–[9], respectively. The point cloud-based methods [1], [2] generally adopts propagation strategy to sequentially densify the sparse key points set, and hence difficult to be fully parallelized. Volumetric methods [3], [4] discretize the 3D space into voxels and then apply photometric measures to determine whether each voxel adheres to the surface, requiring high memory consumption. Comparatively, the depth map-based methods are more flexible and efficient. The depth map-based MVS approaches [5]–[9] decouples the complicated MVS process into a two-stage process comprising *depth map estimation* and *depth map filtering & fusion*. The first stage estimates the per-view depth map and the second stage fuses all the estimated depth maps into the volumetric [10] or the point cloud reconstructions [11]. Although traditional approaches have achieved highly detailed reconstructions under rigid Lambertian textured scenarios, they still suffer from illumination changes, low-textured areas, specular and reflective surfaces, and repeated patterns leading to unreliable pixel correspondences, and hence inaccurate and incomplete reconstruction.

B. Learning-based Multi-View Stereo

Learning-based MVS methods significantly outperform the traditional counterparts in MVS benchmarks [12]–[14]. Pioneer works including SurfaceNet [15] and LSM [16] warp the multi-view image features into the voxel-based cost volumes regularized by the 3D CNNs to regress surface voxels, suffering from the common shortcomings of the volumetric representations and hard to scale up to large-scale scene reconstruction. Comparatively, MVSNet [17] leverages differentiable homography warping to construct the cost volume based on multi-view image features and predefined depth hypotheses and then adopts 3D CNNs to regularize the cost volume and estimate the per-view depth, enabling large-scale scene reconstruction. MVSNet [17] is regarded as the seminal work of the end-to-end learning-based MVS methods. However, it cannot scale up to high-resolution images due to limited memory and high computational cost, simultaneously confront with multi-view matching ambiguity due to heuristic cost volume aggregation strategy ignoring varying significance from different views.

To achieve high-resolution depth estimation, several variants are proposed upon the MVSNet to improve the network scalability to high-resolution images by reducing memory footprint through coarse-to-fine framework [18]–[21] or recurrent neural network (RNN) [22]–[25]. RNN-based methods regularize the cost volume recurrently to trade runtime for memory footprint, leading to slow inference speed despite the processing ability for high-resolution images. In contrast, methods based on the coarse-to-fine framework firstly infer a low-resolution (coarse) depth map based on the predefined depth range with large depth interval, and then gradually narrow the depth range and interval to reduce runtime and memory footprint for achieving high-resolution (fine) depth map estimation. Concurrently, different cost volume aggregation modules are presented to enhance the matching ambiguity by learning the pixel-wise weight [26], patch-wise weight [27], channel-wise weight [28] or voxel-wise weight [21], [24], [29] for cost volume aggregation through a re-weight network. However, the additional re-weight network imposes computational burden and ignores the intrinsic correspondences between multi-view images. In our paper, we propose the LCM scheme to adapt to multi-view scene variation and concurrently alleviate the computational burden. We also enhance the shallow feature information flow via a bottom-up path to tackle the over-smoothing depth estimation, impose end-to-end supervision through adapted focal loss to achieve continuous depth estimation, and adopt the coarse-to-fine framework to speed up the inference time while keeping an efficient memory footprint.

II. Methodology

Network Architecture Please see clearer version of our network architecture in Fig. 4.

A. Normalized Matching Score

The computational procedure of the matching score $\{S_i\}_{i=1}^{N}$ between the i_{th} source image $\{\mathbf{I}\}_{i=1}^{N}$ and the reference image $\mathbf{I}_0$ is detailed in the following algorithm, $\{\mathbf{p}_{ij} \in \mathbb{R}^{3\times1}, j \in \{0, 1, ..., n_i - 1\}\}_{i=1}^{N}$ is the inhomogeneous coordinates of the common 3D points visible in both reference image and i_{th} source image, where n_i is the total number of points triangulated by reference view and i_{th} source view. $\{\mathbf{c}_0, \mathbf{c}_i\} \in \mathbb{R}^{3\times1}$ is the inhomogeneous coordinates of the reference-view and i_{th} source-view camera center, respectively, and θ_j is the baseline angle of $\mathbf{p}_{ij}$. The matching score S_i is accumulated based on a piecewise gaussian function favoring the particular baseline angle θ_0. We then set the normalized matching score $\frac{S_i}{\sum_{i=1}^{N} S_i}$ as the i_{th} source-view significance to make the network adaptive to the input scene variation.

B. Depth Estimation Overview

Recall that our proposed LCM-MVSNet is a three-stage network with three resolution levels including coarse level ($l = 2$), middle level ($l = 1$), and fine level ($l = 0$). We conduct the depth estimation from the coarse level to the fine level to estimate the depth map $\mathbf{D}_{est,0}$ for the reference image $\mathbf{I}_0$.

For the coarsest level $l = 2$, we uniformly sample $(M_2 + 1)$ parallel depth planes (GREEN lines in Fig. 1) from the depth range $[d_{min,2}, d_{max,2}]$ measured at the reference view:

Algorithm 1: Matching Score Computation

Input : $\{\mathbf{p}_{ij} \in \mathbb{R}^{3\times1}, j \in \{0,1,...,n_i-1\}\}_{i=1}^{N}$; Reference-view and source-view camera extrinsics $\mathbf{R}_0 \in \mathbb{R}^{3\times3}$, $\mathbf{t}_0 \in \mathbb{R}^{3\times1}$, $\{\mathbf{R}_i\}_{i=1}^{N} \in \mathbb{R}^{3\times3}$, $\{\mathbf{t}_i\}_{i=1}^{N} \in \mathbb{R}^{3\times1}$.

Output : Matching score $\{S_i\}_{i=1}^{N}$ between $\mathbf{I}_0$ and $\{\mathbf{I}_i\}_{i=1}^{N}$.

Initialization: Favoring baseline angle $\theta_0 = 5°$; Standard deviation of the piecewise gaussian function $\sigma_1 = 1$ and $\sigma_2 = 10$; Matching score $S_i = 0$.

Referece-view camera center $\mathbf{c}_0 = -\mathbf{R}_0^T\mathbf{t}_0$;

for $i = 1$ **to** N **do**

  Source-view camera center $\mathbf{c}_i = -\mathbf{R}_i^T\mathbf{t}_i$;

  for $j = 0$ **to** $n_i - 1$ **do**

    $\theta_j = \frac{180°}{\pi}\arccos\frac{(\mathbf{c}_0-\mathbf{p}_{ij})\cdot(\mathbf{c}_i-\mathbf{p}_{ij})}{||\mathbf{c}_0-\mathbf{p}_{ij}||_2||\mathbf{c}_i-\mathbf{p}_{ij}||_2}$

    if $\theta_j \leq \theta_0$ **then**

      $S_i = S_i + \exp(-\frac{(\theta_j-\theta_0^2)}{2\sigma_1^2})$

    else

      $S_i = S_i + \exp(-\frac{(\theta_j-\theta_0^2)}{2\sigma_2^2})$

return S_i

$$\mathbf{D}_{max,2} = d_{max,2} \tag{1}$$

$$\mathbf{D}_{min,2} = d_{min,2} \tag{2}$$

$$\mathbf{D}_{m,2} = d_{min,2} + m \cdot \frac{d_{max,2}-d_{min,2}}{M_2},\ m \in \{0,1,...,M_2\} \tag{3}$$

Where $\mathbf{D}_{max,2}$, $\mathbf{D}_{min,2}$, $\mathbf{D}_{m,2}$ stands for the maximum, minimum, and sampled depth plane hypotheses. Notably, we conduct depth hypotheses sampling to discretize the 3D space into (M_2+1) parallel depth planes along the depth direction, this does not mean that we assume the object in the reference image $\mathbf{I}_0$ is at the same depth as each pixel in $\mathbf{I}_0$ has (M_2+1) depth candidates. Specifically, based on the depth plane hypotheses, we construct the cost volume and conduct the depth estimation to get the depth map estimation $\mathbf{D}_{est,2}$ (RED curve in Fig. 1), where each pixel is assigned its own optimal depth value. For each non-planar object (a group of pixels) within the reference image, its depth estimation (per-pixel depth values) does not lie in the same depth plane.

For the middle level $l = 1$, we conduct depth range refinement by utilizing the depth map estimation $\mathbf{D}_{est,2}$ from the coarse level $l = 2$ to derive refined depth hypotheses. Specifically, we center the depth range of the middle level at the $\mathbf{D}_{est,2}$ and concurrently reduce the depth interval I_1 and total number of depth hypotheses (M_1+1) at middle level $l = 1$. The process can be formulated as follows:

$$\mathbf{D}_{max,1} = \mathbf{D}_{est,2} + \frac{(M_1+1)}{2} \cdot I_1 \tag{4}$$

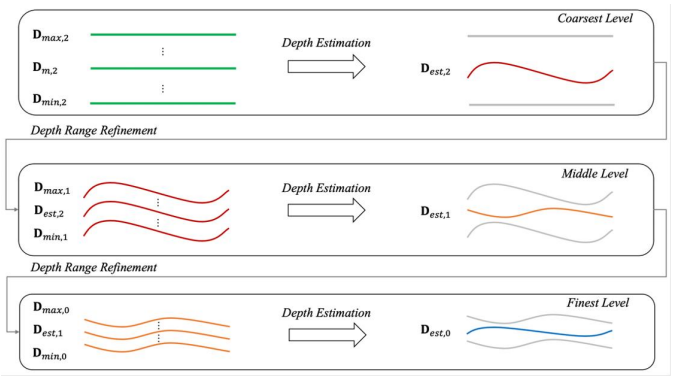

Fig. 1: Overview of the coarse-to-fine depth estimation strategy.

$$\mathbf{D}_{min,1} = \mathbf{D}_{est,2} - \frac{(M_1+1)}{2} \cdot I_1 \tag{5}$$

$$I_1 = r_1 \cdot I_2,\ r_1 < 1 \tag{6}$$

$$M_1 = \rho_1 \cdot M_2,\ \rho_1 < 1 \tag{7}$$

$$\mathbf{D}_{m,1} = \mathbf{D}_{min,1} + m \cdot \frac{\mathbf{D}_{max,1}-\mathbf{D}_{min,1}}{M_1},\ m \in \{0,1,...,M_1\} \tag{8}$$

Where $\mathbf{D}_{max,1}$, $\mathbf{D}_{min,1}$, $\mathbf{D}_{m,1}$ stands for the maximum, minimum, and sampled depth hypotheses, respectively. r_1, ρ_1 is the reduction factor of the depth interval and number of depth hypotheses, respectively. We use RED curves to represent the refined depth hypotheses $\mathbf{D}_{m,1}$ at the middle level $l = 1$. We then construct the cost volume and conduct the depth estimation to get the depth map estimation $\mathbf{D}_{est,1}$ (ORANGE curve) at the middle level $l = 1$. For the fine level $l = 0$, we can repeat the same process to get the final depth map estimation $\mathbf{D}_{est,0}$ (BLUE curve) for the reference image $\mathbf{I}_0$.

C. Cost Volume Construction and Continuous Depth Estimation

After presenting the overview of depth estimation strategy, this section demonstrates the process of cost volume construction and how we achieve continuous depth estimation. For the ease of demonstration, we demonstrate the cost volume construction process for the coarse level $l = 2$, and we omit level ordinal l for clarity.

As aforementioned, we first discretize the 3D space into $(M+1)$ depth values by uniformly sampling a family of fronto-parallel planes (GREEN rectangle in Fig. 2) $\pi_m = [n_0^T, d_m]^T$:

$$d_m = d_{min} + m \cdot \frac{\mathbf{d}_{max}-\mathbf{d}_{min}}{M},\ m \in \{0,1,...,M\} \tag{9}$$

As multi-view stereo is essentially equivalent to solving the pixel correspondences across multi-view images, we adopt the homography to establish the pairwise pixel correspondence

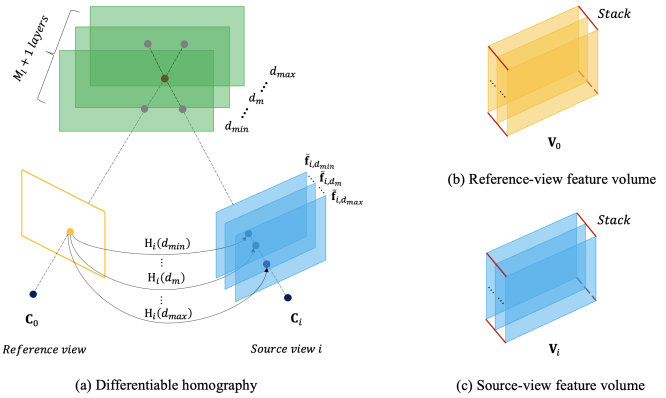

Fig. 2: Construction of the reference-view and source-view feature volume: (a) Differentiable homography; (b) Reference-view feature volume; (c) Source-view feature volume.

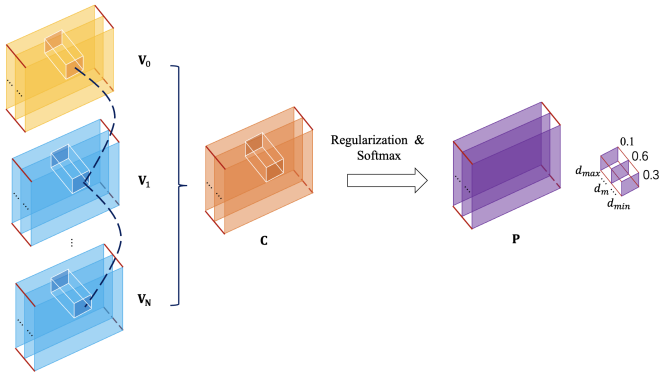

Fig. 3: Cost volume aggregation and regularization

TABLE I: Comparison Between the Evaluation Metrics on Different Benchmarks

Benchmark	Evaluation Metrics	Indication
DTU	Accuracy ↓, Completeness ↓, and Overall Score ↓	Reconstruction Performance
Tanks and Temples	F-score ↑	Reconstruction Performance
BlendedMVS	EPE ↓, e_1 ↓, e_2 ↓	Depth Inference Performance

between the reference-view feature map $\mathbf{f}_0$ and the source-view feature map $\mathbf{f}_i$, $i \in \{1, 2, ..., N\}$, where N stands for the total number of source-view feature maps. Specifically, for each depth value d_m, we conduct feature warping based on homography as follows:

Each d_m determines a homography between the reference view and ith source view:

$$\mathbf{H}_i(d_m) = \mathbf{K}_i\mathbf{R}_i\left(\mathbf{I} - \frac{(\mathbf{C}_0 - \mathbf{C}_i)\mathbf{n}_0^T}{d_m}\right)\mathbf{R}_0^T\mathbf{K}_0^{-1} \quad (10)$$

Then, the coordinate correspondence (image coordinate mapping) between the reference view and ith source view can be established:

$$\mathbf{x}_i = \mathbf{H}_i(d_m) \cdot \mathbf{x}_0 \quad (11)$$

To establish feature correspondence, we then adopt differentiable bilinear interpolation to sample pixels from the source-view feature map $\mathbf{f}_i$, where $\mathbf{x}_i$ specifies pixel location. After interpolation, we can derive the warped source-view feature map $\widetilde{\mathbf{f}}_{i,d_m}$ corresponds to reference-view feature map $\mathbf{f}_0$ under the depth d_m.

We repeat the above process for each depth value d_m from $(M+1)$ depth values, i.e., we conduct the homography transformation and feature warping $(M+1)$ times to get $(M+1)$ warped feature map $\widetilde{\mathbf{f}}_{i,d_m}$ (BLUE rectangle in Fig. 2) aligned to the reference feature map $\mathbf{f}_0$ (YELLOW rectangle in Fig. 2). After feature warping, we stack $(M+1)$ warped feature maps along the depth dimension to get the source-view feature volume $\mathbf{V}_i$ (BLUE volume in Fig. 2) for the ith source view. We stack the reference feature map $\mathbf{f}_0$ $(M+1)$ times along the depth dimension to get the reference-view feature volume $\mathbf{V}_0$ (YELLOW volume in Fig. 2).

We then adopt the proposed learnable cost metric to adaptively fuse the multi-view feature volumes $\{\mathbf{V}_i\}_{i=0}^N$ into the cost volume $\mathbf{C}$ (ORANGE volume) which measures the multi-view feature matching similarity, i.e., per-pixel volume in $\mathbf{C}$ represents multi-view feature matching similarity of the pixel along the depth dimension.

We adopt the 3D CNN to regularize the cost volume and use softmax operation to transform the matching similarity into the probability, and hence we get probability volume $\mathbf{P}$ (PURPLE volume in Fig. 3), where per-pixel volume in $\mathbf{P}$ stands for the depth probabilities of the pixel along the depth dimension, where the depth value corresponds to the maximum probability will be chosen (e.g., in Fig. 3, d_m will be chosen as it has maximum probability 0.6) and assigned to the pixel. Notably, here the depth estimation is discrete as we can only get the optimal depth from the $(M+1)$ depth hypotheses. To get continuous depth estimation, we further refine the discrete depth estimation with the estimated bias between the target depth and discrete depth (See Subsection II. C of the paper).

III. Multi-View Stereo Experiments

A. Datasets

DTU Dataset is a large-scale MVS dataset containing over 100 indoor object-centric scenes. Each scene contains multi-view images captured from 49 or 64 fixed camera positions under 7 different lighting conditions, as well as reference structured light scan serving as the ground-truth point cloud for evaluating the reconstruction quality. We follow the same training (79 scans), validation (18 scans), and evaluation (22 scans) split settings as MVSNet.

BlendedMVS Dataset is a large-scale synthetic dataset for MVS training, fine-tune, and evaluation, and comprises 113 complex objects and scenes, splitting into 106 training scans and 7 validation scans.

Tanks and Temples Dataset serves as a public MVS benchmark and comprises realistic indoor and outdoor scenes. The dataset is divided into the *intermediate* and *advanced* set, consisting of 8 and 6 scenes, respectively, where different scenes differentiate in the scene scale, exposure condition, and surface reflection.

B. Evaluation Metrics

We follow the standard evaluation procedure to evaluate our approach on different benchmark datasets respectively. We systematically compare the evaluation metrics in Table I.

1) DTU Dataset: DTU dataset provides official evaluation protocol to quantify the reconstruction error of the multi-view stereo method in terms of accuracy and completeness [30], [31]. For the target scene, denote the reconstructed point cloud as $\mathcal{R}$ and the ground-truth point cloud as $\mathcal{G}$, then the accuracy and completeness of the point cloud reconstruction are defined as follows:

Accuracy is measured as the distance from $\mathcal{R}$ to $\mathcal{G}$, representing the quality of the reconstructed points, i.e., how close $\mathcal{R}$ lies to $\mathcal{G}$. Specifically, for every point $\mathbf{r}$ in $\mathcal{R}$, compute its Euclidean distance to the closest point in $\mathcal{G}$ and iterate this computation over all points in $\mathcal{R}$ to get the distance distribution, from which the mean value is calculated as accuracy.

$$e_{\mathbf{r}\to\mathcal{G}} = \min_{\mathbf{g}\in\mathcal{G}} \|\mathbf{r}-\mathbf{g}\|_2 \tag{12}$$

$$Accuracy = \frac{1}{|\mathcal{R}|}\sum_{\mathbf{r}\in\mathcal{R}} [e_{\mathbf{r}\to\mathcal{G}} < d] e_{\mathbf{r}\to\mathcal{G}} \tag{13}$$

Where $\|\cdot\|_2$ denotes the Euclidean distance, $|\cdot|$ is the number of points, $[\cdot]$ represents the Iverson bracket, and d stands for the outlier rejection threshold.

Completeness is measured as the distance from $\mathcal{G}$ to $\mathcal{R}$, representing the extent to which the ground truth $\mathcal{G}$ is restored. Specifically, for every point $\mathbf{g}$ in $\mathcal{G}$, compute its Euclidean distance to the closest point in $\mathcal{R}$ and iterate this computation over all points in $\mathcal{G}$ to get the distance distribution, from which the mean value is derived as completeness.

$$e_{\mathbf{g}\to\mathcal{R}} = \min_{\mathbf{r}\in\mathcal{R}} \|\mathbf{g}-\mathbf{r}\|_2 \tag{14}$$

$$Completeness = \frac{1}{|\mathcal{G}|}\sum_{\mathbf{g}\in\mathcal{G}} [e_{\mathbf{g}\to\mathcal{R}} < d] e_{\mathbf{g}\to\mathcal{R}} \tag{15}$$

Trade-off There exists a trade-off between the reconstruction accuracy and completeness, where the accuracy can be maximized with sparse but precisely localized point cloud and the completeness can be maximized with dense point cloud covering the whole space. That is, the MVS method may achieve high accuracy but with low completeness or high completeness but with low accuracy.

Notably, the above accuracy and completeness are defined for the single scene. For quantitative benchmarking on the *DTU evaluation set* (Table I of the manuscript), the accuracy and completeness are the mean accuracy and mean completeness over 22 scenes. To avoid imbalance between accuracy and completeness, overall score takes the average of the mean accuracy and mean completeness to measure the overall reconstruction performance.

2) Tanks and Temples Dataset: Tanks and Temples [13] defines the precision and recall to quantify the reconstruction accuracy and completeness, respectively. The definition of precision and recall is similar to the accuracy and completeness defined in the *DTU* dataset. The main difference is that *DTU* adopts mean error distance in millimeter (mm) but the Tanks and Temples adopts mean error distance in percentage (%). The precision and recall are formulated as follows:

$$e_{\mathbf{r}\to\mathcal{G}} = \min_{\mathbf{g}\in\mathcal{G}} \|\mathbf{r}-\mathbf{g}\|_2 \tag{16}$$

$$Precision = \frac{100}{|\mathcal{R}|}\sum_{\mathbf{r}\in\mathcal{R}} [e_{\mathbf{r}\to\mathcal{G}} < d] \tag{17}$$

$$e_{\mathbf{g}\to\mathcal{R}} = \min_{\mathbf{r}\in\mathcal{R}} \|\mathbf{g}-\mathbf{r}\|_2 \tag{18}$$

$$Recall = \frac{100}{|\mathcal{G}|}\sum_{\mathbf{g}\in\mathcal{G}} [e_{\mathbf{g}\to\mathcal{R}} < d] \tag{19}$$

To overcome the trade-off effect between precision and recall, the *Tanks and Temples* [13] adopts *F-score* to present the summary measure of the reconstruction performance. A high *F-score* can be achieved only if the reconstruction is both accurate and complete.

$$F = \frac{2\cdot Precision\cdot Recall}{Precision + Recall} \tag{20}$$

For benchmarking on the *Tanks and Temples* dataset, the *Mean F-score* on both the *intermediate set* and *advanced set* are reported. For *intermediate set*, the *F-score* of the following 8 scenes are reported: Family (Fam.), Francis (Fra.), Horse (Hor.), Lighthouse (Lig.), M60, Panther (Pan.), Playground (Pla.), Train (Tra). For *advanced set*, the *F-score* of the following 6 scenes are reported: Auditorium (Aud.), Ballroom (Bal.), Courtroom (Cou.), Museum (Mus.), Playground (Pla.), Temple (Tem.). Finally, *Mean F-score* is computed as the arithmetic mean of the F-score of all scenes on the *intermediate set* and *advanced set* respectively.

3) BlendedMVS Dataset: Different from the previous two datasets concentrating on evaluating the reconstruction performance, the *BlendedMVS* [32] dataset adopts the following metrics to focus on measuring the depth estimation quality: *end point error* (EPE), the mean absolute error between the predicted and ground-truth depth map; e_1 and e_3 represents the percentage of pixels in predicted depth map with absolute errors greater than 1 and 3 in comparison to the ground-truth depth map, respectively.

C. Implementation Details

Training We train our LCM-MVSNet on the *DTU training set* and then evaluate it on the *DTU evaluation set* to benchmark the performance of our network. As the original *DTU* dataset only contains ground-truth point clouds, for end-to-end network training, we follow the common practices to acquire ground-truth depth maps through screened Possion surface reconstruction followed by depth rendering. Our network is defined as a three-stage coarse-to-fine network after balancing the accuracy and the efficiency. The depth hypotheses are

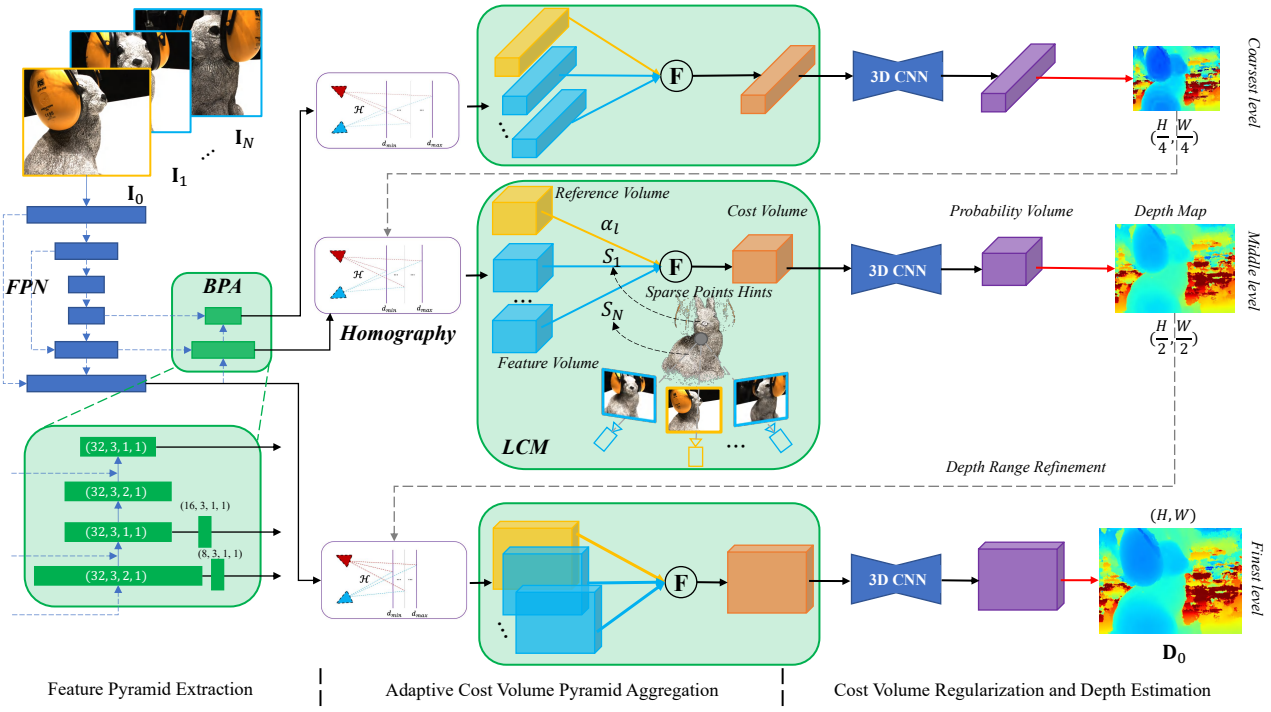

Fig. 4: **Illustration of LCM-MVSNet network**. Our network is a coarse-to-fine network taking as input the multi-view images $\{\mathbf{I}_i\}_{i=0}^{N}$ and inferring depth map $\mathbf{D}_0$. From left to right, we first utilize the *feature pyramid extraction network* to extract multi-scale feature pyramid, encoded via *differentiable homography warping* to construct reference volume pyramid and feature volume pyramids, adaptively fused by the proposed *LCM module* to aggregate the cost volume pyramid for regularizing and inferring the depth map $\mathbf{D}_0$ in a coarse-to-fine manner.

uniformly sampled from $425mm$ to $935mm$. From the coarsest stage to the finest stage, the number of depth hypotheses is set to 48, 32, and 8. Accordingly, the depth interval of each stage is set to 4, 2, 1 times of the depth interval at the coarsest stage. We set the number of input views to 5 and the input image resolution to 640×512. Note that the previous settings are consistent with the state-of-the-art methods for a fair comparison. We implement the network by PyTorch and optimize it with the Adam optimizer ($\beta_1 = 0.9$, $\beta_2 = 0.999$) for 60 epochs on two NVIDIA RTX 3090Ti GPUs with batch size 2 on each GPU. The cosine learning rate scheduler with an initial learning rate of 0.001 is adopted for decaying the learning rate.

Evaluation For benchmarking on the *DTU evaluation set*, we resize the input image resolution to 1152×864 and set the number of input views to 5 for depth map estimation. The estimated depth maps are then filtered and fused into the point cloud to evaluate the mean error distance (lower the better) with respect to (w.r.t.) the ground-truth point cloud. In our experiments, we set the probability threshold τ as 0.1 to discard depth outliers and set the number of consistent views N_c as 3 to reduce the depth inconsistency.

For benchmarking on the *Tanks and Temples*, as *DTU* dataset only contains indoor scenes with fixed camera positions, we finetune our model on the *BlendedMVS* training set with complex scene variations and diverse camera trajectories to improve the generalization ability. For finetuning, we set the original input image resolution as 768×576, and the number of input views as 7. For benchmarking, we set the number of input views as 11 to estimate depth maps, subsequently filtered and fused into the final point cloud to compute the *F-score* (higher the better) w.r.t. the ground-truth point cloud.

For benchmarking on the *BlendedMVS* validation set, we adopt the original input image resolution 768×576 and set the number of input views to 5 for all methods to ensure a fair comparison of the depth estimation quality. The EPE, e_1, and e_3 (lower the better) are recorded.

D. Ablation Study

Bottom-up Path Augmentation The depth estimation may suffer from over-smoothing around the object boundaries due to the lack of shallow feature information containing low-level features such as local textures and edges. As shown in Fig. 5, the baseline model confronts with over-smoothing depth estimation around the object boundary with non-Lambertian low-textured surface. To tackle this issue, we introduce a bottom-up path augmentation (BPA) with negligible parameters increasing to shorten the propagation of shallow information, shown to be conducive to depth estimation in Fig. 5. The figure also demonstrates that the proposed LCM scheme enhances the accuracy and completeness of depth estimation and subsequent reconstruction.

Number of Input Views N, Image Resolution $W \times H$, Probability Threshold τ and Number of Consistent Views N_c for Depth Fusion Systematic ablation experiments are

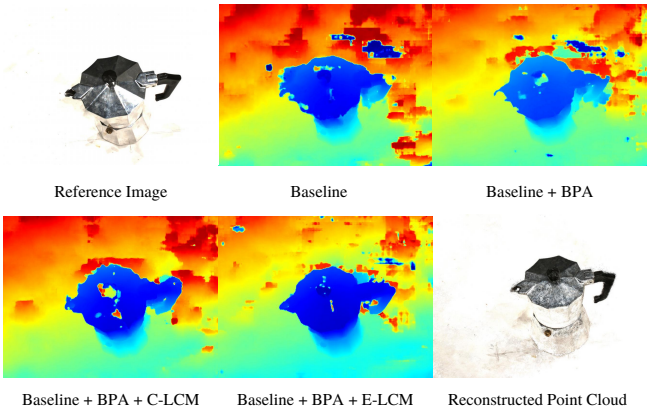

Fig. 5: Qualitative comparison of depth map estimations on the *Scan77* of the *DTU evaluation set*. The BPA module and LCM scheme can improve the accuracy, completeness, and continuity of the depth estimation under non-Lambertian low-textured surface.

conducted to analyze their influence on the reconstruction quality based on *Model D* in Table IV of the paper. As shown in the first part of the Table II, the model performs continuously better in terms of all three metrics with increasing N within 7 and achieves the best *accuracy* and *overall score* when $N = 7$. However, large N ($N = 11$) leads to the loss in the reconstruction *accuracy, completeness* and *overall score*, attributed to the redundant information induced by the excess views. We then change τ from 0.1 to 0.5 to investigate its impact as shown in the second part of the Table II, illustrating that smaller τ results in worse *accuracy*, better *completeness* and *overall score*, and a larger value leads to the opposite result. In particular, the model achieves the best *completeness* and *overall score* when $\tau = 0.1$. Moreover, we increase N_c from 2 to 6 as shown in the third part of the Table II and find that larger N_c leads to better *accuracy* and worse *completeness*, where the best *accuracy* and *completeness* is obtained when $N_c = 6$ and $N_c = 2$ respectively. The best *overall score* is acquired when $N_c = 3$ as a balanced choice. Furthermore, we report the impact of input image resolution in the fourth part of the Table II, where the best *overall score* is obtained when $W \times H = 1152 \times 864$. In summary, we set $N = 5$ & $7, \tau = 0.1$ & $0.3, N_c = 3$ *to* $6, W \times H = 1152 \times 864$ to achieve the state-of-the-art performance and benchmark the results on the *DTU evaluation set* as shown in the fifth part of the Table II and Table I of the paper.

E. Benchmarking Results Analysis of Tanks and Temples

As shown in Table II of the paper, our method cannot achieve optimal *F-score* when reconstructing Horse, Lighthouse, Playground, and Train. For scene Horse and Train, our method is inferior to TransMVSNet [20]. For scene Lighthouse and Playground, our method is inferior to AttMVS [28]. Recall that learning-based multi-view stereo (MVS) methods decouple the MVS process into a two-stage process including: learning-based depth estimation (multi-view images to multi-view depth maps) and depth map fusion (multi-view depth

TABLE II: Ablation Experiments on N, τ, N_c, and $W \times H$

N	τ	N_c	$W \times H$	Mean Error Distance		
				Acc. ↓ (mm)	Comp. ↓ (mm)	Overall ↓ (mm)
3				0.389	0.349	0.369
4				0.369	0.290	0.330
5				0.358	0.275	0.317
6				0.359(+0.0003)	**0.271**(+0.0002)	0.315(+0.0003)
7	0.3	3	1152 × 864	**0.356**(+0.0001)	0.273(+0.0001)	**0.315**(-0.0004)
8				0.356	0.276	0.317
9				0.357	0.280	0.319
10				0.358	0.286	0.322
11				0.359	0.292	0.326
	0.1			0.368(-0.0002)	**0.263**(-0.0001)	**0.315**(+0.0003)
	0.2			0.366(-0.0002)	0.265(-0.0001)	**0.315**(+0.0003)
5	0.3	3	1152 × 864	0.358	0.275	0.317
	0.4			0.348	0.294	0.321
	0.5			**0.332**	0.334	0.333
		2		0.431	**0.231**	0.331
		3		0.358	0.275	**0.317**
5	0.3	4	1152 × 864	0.315	0.341	0.328
		5		0.285	0.427	0.356
		6		**0.262**	0.539	0.401
			768 × 576	0.386	0.288	0.337(-0.0001)
5	0.3	3	1152 × 864	**0.358**	**0.275**	**0.317**
			1536 × 1152	0.363	0.312	0.338(-0.0005)
5	0.1	3	1152 × 864	0.368(-0.0002)	0.263(-0.0001)	0.315(+0.0003)
7	0.1	3	1152 × 864	0.364(+0.0002)	**0.262**(+0.0002)	**0.313**(+0.0002)
7	0.1	4	1152 × 864	0.317	0.323	0.320
7	0.1	5	1152 × 864	0.287	0.397	0.342
7	0.1	6	1152 × 864	**0.265**	0.492	0.379

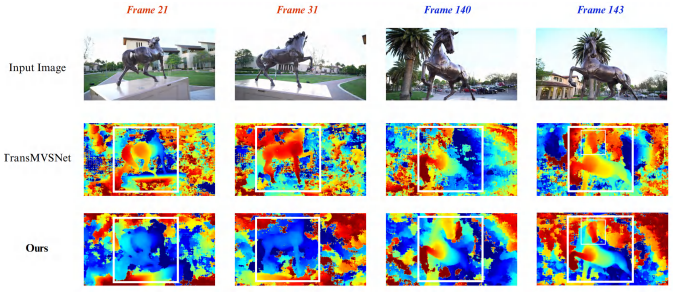

Fig. 6: Qualitative comparison of depth estimation between our method and TransMVSNet on scene Horse of the intermediate set. For the Horse with specular surfaces, our method is inferior to TransMVSNet under backlight (frames marked in RED) and our method is superior to TransMVSNet without backlight (frames marked in BLUE).

maps to point cloud). To analyze the specific reason, for each scene, we compare our method with others from two aspects: depth estimation performance and point cloud reconstruction performance.

1) Horse: For depth estimation, the qualitative comparison between our method and TransMVSNet is shown in Fig. 6. For the Horse with specular surface, our method can achieve more accurate and complete depth estimation under environment with less backlight (see Frame 140 and Frame 143). Nevertheless, with strong backlight, although our method can infer the depth map by differentiating between foreground and background, the estimated depth value is less accurate com-

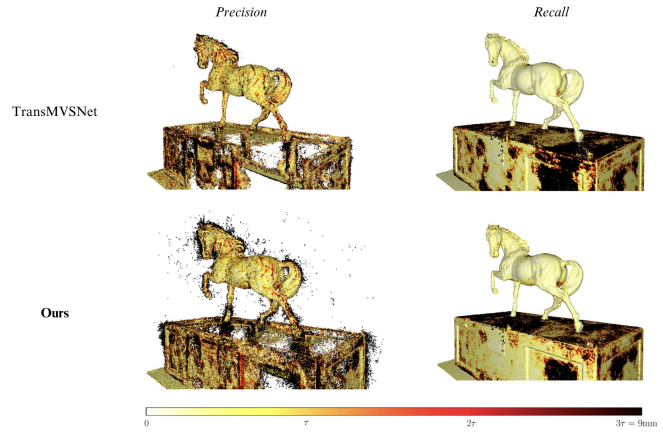

Fig. 7: Visualization of point cloud reconstruction error of our method and TransMVSNet on scene Horse of the intermediate set. τ is the per-scene point distance threshold defined by the benchmark and darker color indicates a larger reconstruction error with respect to τ.

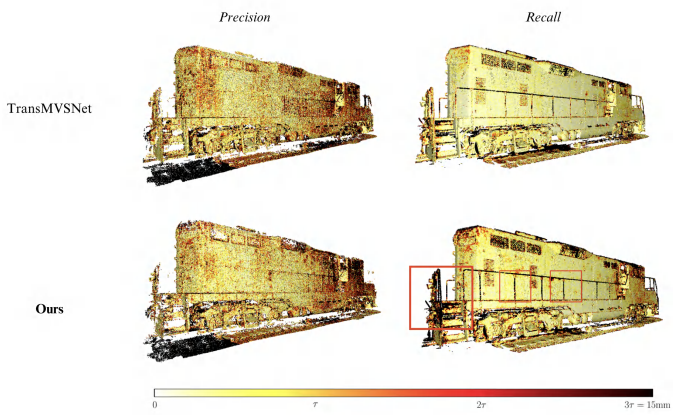

Fig. 8: Visualization of point cloud reconstruction error of our method and TransMVSNet on scene Train of the intermediate set. τ is the per-scene point distance threshold defined by the benchmark and darker color indicates a larger reconstruction error with respect to τ.

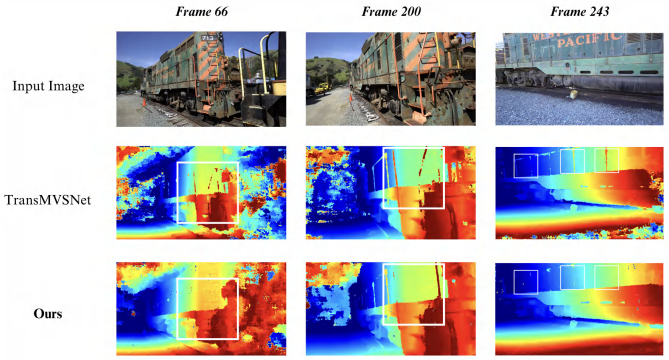

Fig. 9: Qualitative comparison of depth estimation between our method and TransMVSNet on scene Train of the intermediate set. Our method achieves comparable depth estimation completeness w.r.t. TransMVSNet except for the slim structure such as guard bar.

pared to TransMVSNet. For example, for Frame 21 and 31, our method deduces the Horse is far to the camera (rendered in BLUE) while the TransMVSNet accurately deduces the Horse is close to the camera (rendered in RED). TransMVSNet adopts transformer to strengthen the long-range global context aggregation between the images, where Frame 31 can acquire context information from the accompanying images without backlight to achieve more accurate depth estimation [20].

Based on the estimated depth values, the point cloud is reconstructed by projecting the pixels back to the points in the 3D space. As aforementioned, for the Horse under backlight environment, our method produces less accurate depth estimation compared to the TransMVSNet and this results in the misplaced points around the Horse (see precision rendering of our method in Fig. 7), leading to less reconstruction accuracy. The reconstruction completeness of our method is comparable to the TransMVSNet (see recall rendering of our method in Fig. 7). For scene Horse, the comparable reconstruction completeness but less accurate reconstruction accuracy leads to the inferior F-score compared to TransMVSNet.

2) Train: For scene Train, we visualize the reconstruction error in Fig. 8, where our method achieves comparable reconstruction accuracy (precision) but less reconstruction completeness (recall), especially in slim structure such as guard bar (bounded by the red bounding box). This incompleteness is attributed to the less complete depth estimation (see Fig. 9) of our method for the slim structure compared to TransMVSNet, which benefits from the transformer ensuring a global receptive field for the local features [20].

3) Lighthouse and Playground: For scene Lighthouse and Playground, our method is inferior to AttMVS. AttMVS does not release the corresponding code resource and optimal checkpoints (weights) for evaluating on the Tanks and Temples. And hence we only compare the point cloud reconstruction results by visualizing the precision (reconstruction accuracy) and recall (reconstruction completeness) as shown in Fig. 10 and Fig. 11, our method achieves comparable recall but less precision on both scenes when compared to AttMVS, which may benefit from attention-guided regularization module conducive to the scenes with the interested depth range of the captured images being concentrated [28].

In summary, although our method does not achieve optimal *F-score* for the above 4 scenes, it still achieves competitive performance for these scenes compared to the state of the arts. Notably, our method achieves state-of-the-art performance on all the indoor scenes with large depth ranges. Compared to our baseline method (CasMVSNet), our method significantly improves the reconstruction performance, demonstrating the effectiveness of each component of our method.

F. More Reconstruction Results

We apply our *LCM-MVSNet* to reconstruct all scenes of the *DTU evaluation set* (see Fig. 12), the *intermediate set* of the *Tanks and Temples* benchmark (see Fig. 13), the *advanced set* of the *Tanks and Temples* benchmark (see Fig. 14), *BlendedMVS validation set* (see Fig. 15), demonstrating the

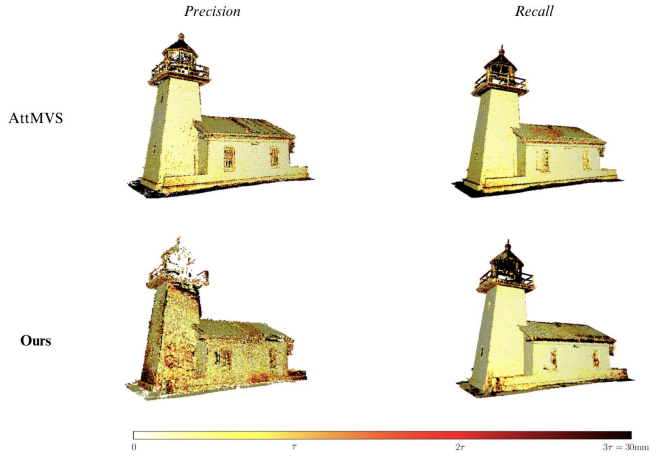

Fig. 10: Visualization of point cloud reconstruction error of our method and TransMVSNet on scene Lighthouse of the intermediate set. τ is the per-scene point distance threshold defined by the benchmark and darker color indicates a larger reconstruction error with respect to τ.

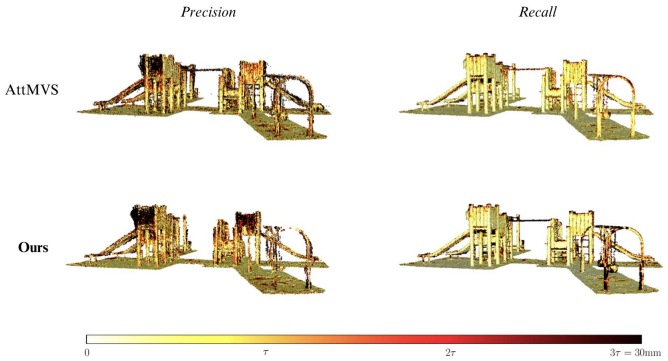

Fig. 11: Visualization of point cloud reconstruction error of our method and TransMVSNet on scene Playground of the intermediate set. τ is the per-scene point distance threshold defined by the benchmark and darker color indicates a larger reconstruction error with respect to τ.

strong robustness and scalability of our method on multi-scale scenes with varying depth ranges in both indoor and outdoor environments.

IV. Real-World Application for UAV-Based Infrastructure Defect Inspection and Localization

Runtime Analysis As shown in Table III, we conduct experiments on 5 real-world scenes with varing depth ranges to analyze the runtime of our method in real-world deployment, demonstrating the scalability and efficiency of our method.

Unmanned Aerial System As shown in Fig. 8 (a) of the paper, three UAVs are used to speed up the image collection for defect detection and reconstruction of the target warehouse [33]–[35]. Each UAV will reach the best region according to the task assignment method. After reaching the best region, viewpoints can be generated based on the building morphology. Furthermore, these viewpoints are regarded as the nodes of a traveling salesman problem to determine the shortest path that travels through all these viewpoints. Finally, the generated path will be executed by each UAV to collect the images for inspection and reconstruction. The multi-UAV-based data collection can speed up the overall image collection process by more than 3 times.

TABLE III: Runtime Analysis on Real-World Reconstruction

Scene	Image Amount	Resolution	Runtime (mins)			
			SfM	Depth Estimation	Depth Map Fusion	Total
Facade	51	1152 × 832	0.555	1.737	0.029	2.321
Village	87	1152 × 832	0.914	3.072	0.052	4.038
Tulou	248	1152 × 640	4.266	2.163	0.124	6.553
Campus	543	1152 × 832	17.316	6.137	0.357	23.810
Urban	826	1152 × 832	30.947	9.820	4.161	44.928

TABLE IV: Comparison with Industrial 3D Reconstruction Solutions on the *Advanced Set* of the *Tanks and Temples* Benchmark

Methods	*F-score* ↑ (%)						
	Mean	*Aud.*	*Bal.*	*Cou.*	*Mus.*	*Pal.*	*Tem.*
VisualSfM + OpenMVS	12.70	7.94	15.21	21.21	19.78	9.10	2.99
MVE	18.28	4.11	12.63	27.93	34.67	13.58	16.79
OpenMVG + OpenMVS	21.85	9.79	22.49	26.54	36.89	14.64	20.76
OpenMVG + MVE	22.93	14.70	26.36	32.48	37.57	3.65	22.84
COLMAP	27.24	16.02	25.23	34.70	41.51	18.05	27.94
Pix4D	25.07	10.83	18.53	33.21	47.37	14.47	26.01
Ours	**38.54**	**27.22**	**44.73**	**39.21**	**53.02**	**32.73**	**34.33**

Defect Detection Dataset Due to the lack of publicly available defect dataset annotated in bounding-box level, we establish a high-resolution defect detection dataset comprising over 5,500 visible images. The dataset covers three important defect classes: crack, spalling, and moisture. We split 72% of images for training, 8% for validation, and the remaining 20% for robustness testing.

References

[1] Y. Furukawa and J. Ponce, "Accurate, dense, and robust multiview stereopsis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 8, pp. 1362–1376, 2010.

[2] M. Lhuillier and L. Quan, "A quasi-dense approach to surface reconstruction from uncalibrated images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 3, pp. 418–433, 2005.

[3] K. N. Kutulakos and S. M. Seitz, "A theory of shape by space carving," *International Journal of Computer Vision*, vol. 38, pp. 199–218, 2000.

[4] S. M. Seitz and C. R. Dyer, "Photorealistic scene reconstruction by voxel coloring," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, 1997, pp. 1067–1073.

[5] E. Tola, C. Strecha, and P. Fua, "Efficient large-scale multi-view stereo for ultra high-resolution image sets," *Machine Vision and Applications*, vol. 23, no. 5, pp. 903–920, 2012.

[6] N. D. Campbell, G. Vogiatzis, C. Hernández, and R. Cipolla, "Using multiple hypotheses to improve depth-maps for multi-view stereo," in *European Conference on Computer Vision*. Springer, 2008, pp. 766–779.

[7] S. Galliani, K. Lasinger, and K. Schindler, "Massively parallel multiview stereopsis by surface normal diffusion," in *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 873–881.

[8] J. L. Schönberger, E. Zheng, M. Pollefeys, and J.-M. Frahm, "Pixelwise view selection for unstructured multi-view stereo," in *European Conference on Computer Vision (ECCV)*, 2016.

[9] Q. Xu and W. Tao, "Multi-scale geometric consistency guided multi-view stereo," *Computer Vision and Pattern Recognition (CVPR)*, 2019.

[10] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon, "Kinectfusion: Real-time dense surface mapping and tracking," in *2011 10th IEEE International Symposium on Mixed and Augmented Reality*, 2011, pp. 127–136.

Fig. 12: All point cloud reconstruction results on the *DTU evaluation set*

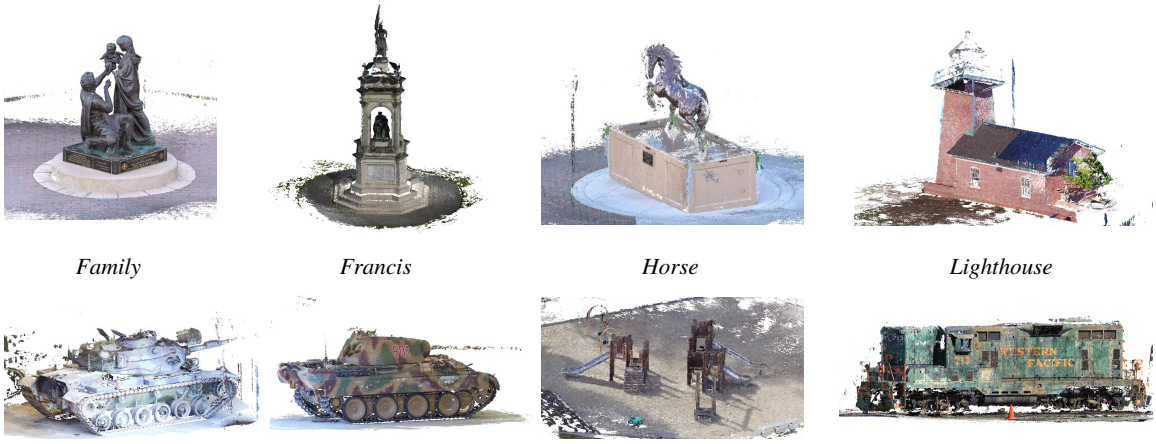

Fig. 13: All point cloud reconstruction results on the *intermediate set* of the *Tanks and Temples* benchmark.

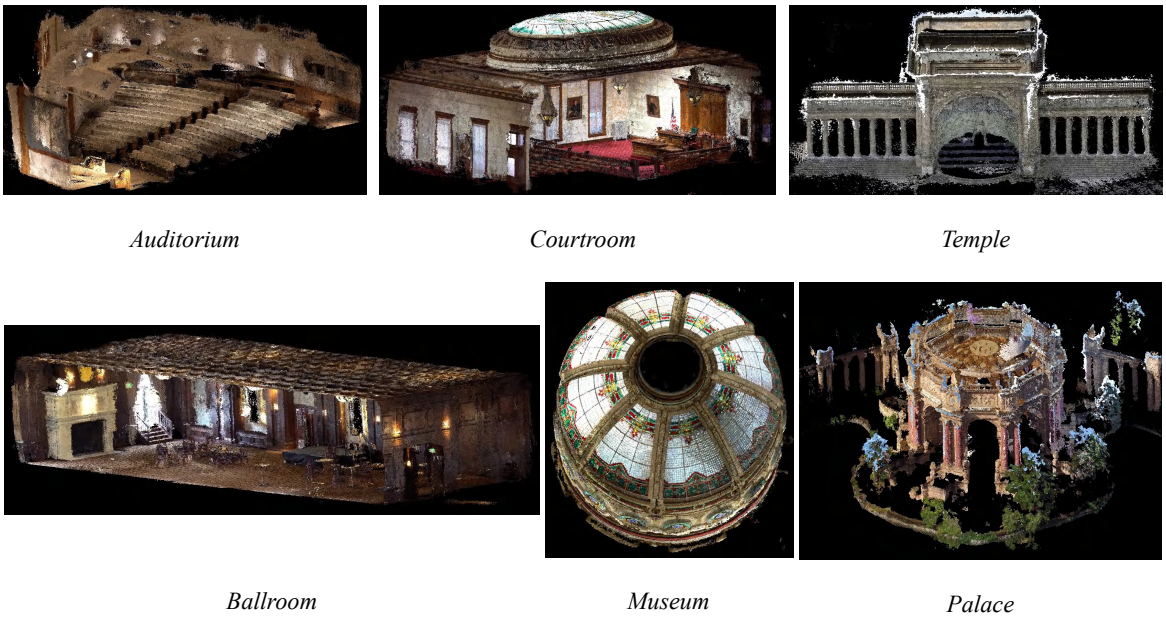

Fig. 14: All point cloud reconstruction results on the *advanced set* of the *Tanks and Temples* benchmark.

Fig. 15: All point cloud reconstruction results of the *BlendedMVS validation set*.

[11] S. Galliani, K. Lasinger, and K. Schindler, "Massively parallel multiview stereopsis by surface normal diffusion," in *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 873–881.

[12] H. Aanæs, R. R. Jensen, G. Vogiatzis, E. Tola, and A. B. Dahl, "Large-scale data for multiple-view stereopsis," *International Journal of Computer Vision*, vol. 120, no. 2, pp. 153–168, 2016.

[13] A. Knapitsch, J. Park, Q.-Y. Zhou, and V. Koltun, "Tanks and temples: Benchmarking large-scale scene reconstruction," *ACM Transactions on Graphics (ToG)*, vol. 36, no. 4, pp. 1–13, 2017.

[14] T. Schops, J. L. Schonberger, S. Galliani, T. Sattler, K. Schindler, M. Pollefeys, and A. Geiger, "A multi-view stereo benchmark with high-resolution images and multi-camera videos," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3260–3269.

[15] M. Ji, J. Gall, H. Zheng, Y. Liu, and L. Fang, "Surfacenet: An end-to-end 3d neural network for multiview stereopsis," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2326–2334.

[16] A. Kar, C. Häne, and J. Malik, "Learning a multi-view stereo machine," *Advances in neural information processing systems*, vol. 30, 2017.

[17] Y. Yao, Z. Luo, S. Li, T. Fang, and L. Quan, "Mvsnet: Depth inference for unstructured multi-view stereo," in *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

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

[19] K. T. Giang, S. Song, and S. Jo, "Curvature-guided dynamic scale networks for multi-view stereo," in *International Conference on Learning Representations*, 2022.

[20] Y. Ding, W. Yuan, Q. Zhu, H. Zhang, X. Liu, Y. Wang, and X. Liu, "Transmvsnet: Global context-aware multi-view stereo network with transformers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022, pp. 8585–8594.

[21] R. Peng, R. Wang, Z. Wang, Y. Lai, and R. Wang, "Rethinking depth estimation for multi-view stereo: A unified representation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022, pp. 8645–8654.

[22] Y. Yao, Z. Luo, S. Li, T. Shen, T. Fang, and L. Quan, "Recurrent mvsnet for high-resolution multi-view stereo depth inference," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[23] J. Yan, Z. Wei, H. Yi, M. Ding, R. Zhang, Y. Chen, G. Wang, and Y.-W. Tai, "Dense hybrid recurrent multi-view stereo net with dynamic consistency checking," in *European conference on computer vision*. Springer, 2020, pp. 674–689.

[24] Z. Wei, Q. Zhu, C. Min, Y. Chen, and G. Wang, "Aa-rmvsnet: Adaptive aggregation recurrent multi-view stereo network," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021, pp. 6187–6196.

[25] ——, "Bidirectional hybrid lstm based recurrent neural network for multi-view stereo," *IEEE Transactions on Visualization and Computer Graphics*, 2022.

[26] J. Zhang, S. Li, Z. Luo, T. Fang, and Y. Yao, "Vis-mvsnet: Visibility-aware multi-view stereo network," *International Journal of Computer Vision*, pp. 1–16, 2022.

[27] K. Luo, T. Guan, L. Ju, H. Huang, and Y. Luo, "P-mvsnet: Learning patch-wise matching confidence aggregation for multi-view stereo," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[28] K. Luo, T. Guan, L. Ju, Y. Wang, Z. Chen, and Y. Luo, "Attention-aware multi-view stereo," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[29] H. Yi, Z. Wei, M. Ding, R. Zhang, Y. Chen, G. Wang, and Y.-W. Tai, "Pyramid multi-view stereo net with self-adaptive view aggregation," in *European Conference on Computer Vision*. Springer, 2020, pp. 766–782.

[30] R. Jensen, A. Dahl, G. Vogiatzis, E. Tola, and H. Aanæs, "Large scale multi-view stereopsis evaluation," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2014, pp. 406–413.

[31] H. Aanæs, R. R. Jensen, G. Vogiatzis, E. Tola, and A. B. Dahl, "Large-scale data for multiple-view stereopsis," *International Journal of Computer Vision*, pp. 1–16, 2016.

[32] Y. Yao, Z. Luo, S. Li, J. Zhang, Y. Ren, L. Zhou, T. Fang, and L. Quan, "Blendedmvs: A large-scale dataset for generalized multi-view stereo networks," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 1787–1796.

[33] X. Wang, C. Gao, X. Chen, and B. M. Chen, "Fast and secure distributed multi-agent coverage control with an application to infrastructure inspection and reconstruction," in *Proceedings of the 42nd Chinese Control Conference*, July 2023, pp. 5998–6005.

[34] Y. Chen, S. Lai, J. Cui, B. Wang, and B. M. Chen, "Gpu-accelerated incremental euclidean distance transform for online motion planning of mobile robots," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 6894–6901, 2022.

[35] C. Gao, W. Ding, Z. Zhao, and B. M. Chen, "Energy-optimal trajectory-based traveling salesman problem for multi-rotors unmanned aerial vehicle," in *62nd IEEE Conference on Decision and Control*, December 2023.