Non-parametric Depth Distribution Modelling based Depth Inference for Multi-view Stereo

Jiayu Yang[1,2*],   Jose M. Alvarez[2],   Miaomiao Liu[1]

[1]Australian National University, [2]NVIDIA

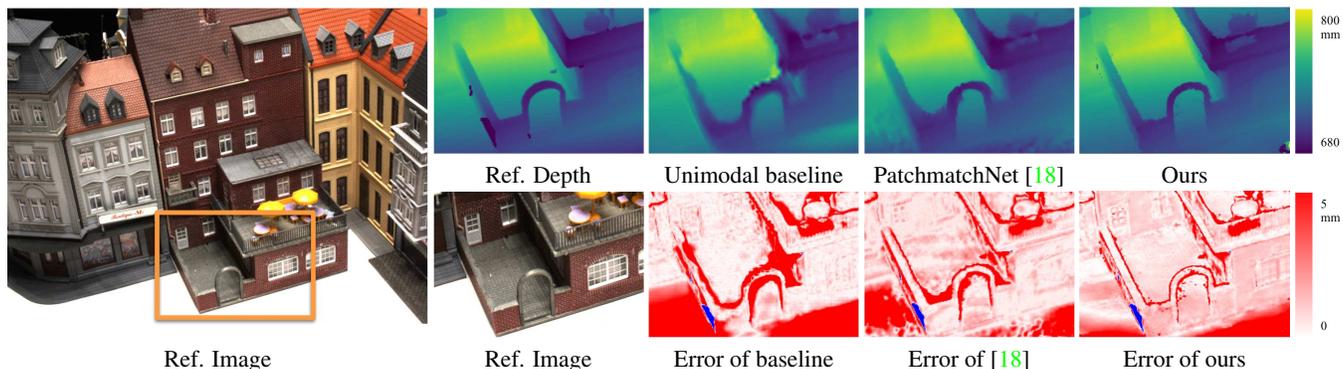{jiayu.yang, miaomiao.liu}@anu.edu.au, josea@nvidia.com

Figure 1. Performance analysis of cascade MVS methods on boundary regions. Compared with the *unimodal baseline* model, and the state-of-the-art patchmatch based approach *PatchmatchNet* [18], our method based on non-parametric depth distribution modeling can achieve more accurate depth estimation results on the boundary regions with abrupt depth changes.

Abstract

Recent cost volume pyramid based deep neural networks have unlocked the potential of efficiently leveraging high-resolution images for depth inference from multi-view stereo. In general, those approaches assume that the depth of each pixel follows a unimodal distribution. Boundary pixels usually follow a multi-modal distribution as they represent different depths; Therefore, the assumption results in an erroneous depth prediction at the coarser level of the cost volume pyramid and can not be corrected in the refinement levels leading to wrong depth predictions. In contrast, we propose constructing the cost volume by non-parametric depth distribution modeling to handle pixels with unimodal and multi-modal distributions. Our approach outputs multiple depth hypotheses at the coarser level to avoid errors in the early stage. As we perform local search around these multiple hypotheses in subsequent levels, our approach does not maintain the rigid depth spatial ordering and, therefore, we introduce a sparse cost aggregation network to derive information within each volume. We evaluate our approach extensively on two benchmark datasets: DTU and Tanks & Temples. Our experimental results show that our model outperforms existing methods by a large margin and achieves superior performance on boundary regions.

Code is available at `https://github.com/NVlabs/NP-CVP-MVSNet`

1. Introduction

Multi-view stereo (MVS) aims to infer the 3D structure, represented by a depth map, of a scene from a set of images captured by a camera from multiple viewpoints. It is a fundamental problem for the computer vision community and has been studied extensively for decades [13].

Recently, learning-based methods using a cost volume pyramid [3, 5, 22] have emerged as the preferred technique to leverage high-resolution images leading to superior performance compared to other learning-based approaches [23,24]. These approaches usually perform a local depth search around an initial depth estimate on a cost volume built at the coarser level. Then, assuming each pixel follows a unimodal distribution, they estimate the depth for each pixel as the expectation of approximated continuous depth distributions by a few depth samples within a predefined range [3, 5, 22]. While these approaches have achieved promising results, they tend to miss small objects and boundary regions with abrupt depth changes, where the

---

*The work is done during an internship at NVIDIA

unimodal distribution assumption does not hold.

As shown in Fig. 1, the multi-scale depth estimation frameworks with unimodal distribution assumption performs badly on boundary pixels. It is mainly because existing multi-scale approaches make an early decision at the coarse level as the depth in this level is represented as a single value based on the unimodal distribution assumption. If the estimated coarse depth is far from the actual depth, the error will be propagated to refinement levels and can not be corrected via local depth search, resulting in incorrect depth prediction.

In this work, we address this problem by explicitly modeling the depth of each pixel at different resolutions using a multi-modal distribution. In particular, in contrast to methods using a mixture density distribution [17], we use a non-parametric distribution to learn the probability of each depth hypothesis along the 3D visual ray. Our approach provides additional flexibility compared to parametric approaches, especially in the coarse-to-fine structure. We then guide the learning process using the depth distribution within its corresponding depth patch at the highest resolution. Given the learned distribution, we build the cost volume for the next level by branching the depth hypotheses using the top K probabilities. Our approach achieves a better ground truth depth covering ratio than the existing approaches; however, it loses the relative spatial relationship due to the pixel-wise depth branching processing. To aggregate the information in our new cost volume structure, we propose a sparse cost aggregation network to preserve the relative spatial relationship. Extensive experiments on several benchmark datasets demonstrate that our approach achieves superior performance, especially on boundary regions. On the DTU dataset, our approach outperforms the current state-of-the-art multi-scale patchmatch based approach Patchmatch-Net [18], yielding up to a 32% lower error on boundary regions.

In summary, the contributions in this paper are as follows.

- We propose a non-parametric depth probability distribution modeling, allowing us to handle pixels with unimodal and multimodal distributions.

- We build a cost volume pyramid by branching the depth samples based on the modeled pixel-wise depth probability distribution.

- We apply a sparse cost aggregation network to process each cost volume to maintain rigid geometric spatial relation in the cost volume and avoid spatial ambiguity.

- Our approach outperforms previous approaches on boundary areas and becomes the new state of the art on the DTU dataset.

2. Related works

Deep Learning-based MVS methods adopt deep CNNs to infer the depth map for each view followed by a separate multiple-view fusion process for building 3D models. These methods allow the network to extract discriminative features encoding global and local information of a scene to obtain robust feature matching for MVS. In particular, Yao *et al.* propose MVSNet [23] to infer a depth map for each view. An essential step in [23] is to build a cost volume based on a plane sweep process followed by multi-scale 3D CNNs for regularization. While effective in depth inference accuracy, its memory requirement is cubic to the image resolution. To allow handling high resolution images, several methods have been proposed to reduce the memory requirement. Recurrent methods [19, 21, 24] build cost volume and aggregate matching cost in a recursive manner using recurrent networks such as GRU [24] or LSTM [19, 21]. However, recurrent methods usually consume a longer run-time trading for reduced space requirement.

Another line of research exploits multi-scale framework for coarse-to-fine depth estimation [2, 3, 5, 18, 20, 22]. Multi-scale methods build a coarse cost volume in lower resolution to estimate a coarse depth map followed by building partial cost volumes in higher resolution for depth refinement. Although multi-scale methods have achieved promising results in both efficiency and accuracy, two major problems remain unsolved: Early decision on coarser levels and spatial ambiguity of partial cost volumes. In this paper, we address the first problem by a non-parametric depth distribution modelling paired with a novel multi-scale depth estimation framework. We address the second problem by a novel sparse cost volume formulation and a sparse cost aggregation network that retain rigid spatial relation.

Multi-modal disparity distribution modeling. Multi-modal disparity probability distribution modeling has been studied recently in tackling the stereo matching problem. In [17], Tosi *et al.* show that the unimodal probability distribution assumption for each pixel leads to inaccurate disparity estimations when the actual disparity follows a multi-modal distribution. They thus propose to model the continuous disparity as a parametric bi-modal distribution. Their proposed approach further improves the disparity estimation performance at the high resolution especially for boundary pixels. While the proposed approach can tackle the multi-modal disparity/depth modelling, it is not trivial to be directly extended to the cost volume pyramid structure to avoid resulting in erroneous depth estimation at the coarse level.

UASNet [9] proposed to build the cascaded cost volume via uncertainty guided depth range estimation and depth sampling. Their goal is to build a cost volume with depth sampling covers the multi-modal depth distribution. In particular, they output 4 disparity maps at each stage. In con-

trast, we do not output depth estimate at the early stage instead of building cost volume based on depth samples with high probability.

3. Method

Let $\mathbf{I}_0 \in \mathbb{R}^{H \times W \times 3}$ be the reference image, where H and W define its dimensions; and let $\{\mathbf{I}_i\}_{i=1}^{N}$ be its N neighboring source images. Assume $\{\mathbf{K}_i, \mathbf{R}_i, \mathbf{t}_i\}_{i=0}^{N}$ are the corresponding camera intrinsic parameters, rotation matrix, and translation vector for all views. Our goal is to infer the depth map $\mathbf{D} \in \mathbb{R}^{H \times W}$ for $\mathbf{I}_0$ from $\{\mathbf{I}_i\}_{i=0}^{N}$.

Figure 2 shows the overall pipeline of our approach to performing depth inference based on a cost volume pyramid built in a coarse-to-fine manner. Unlike existing works, the key idea of our approach is to build the cascaded cost-volume based on the local search around the top K depth hypotheses, which we obtain by modeling the pixel-wise depth probability distribution. Below, we first introduce our the non-parametric depth probability distribution modeling; then the cost volume pyramid built using those non-parametric distributions and the depth map inference process. Finally, we provide details of the loss function.

3.1. Non-parametric depth distribution modelling

Existing methods assume the depth d of a pixel p follows a unimodal probability distribution $\mathbf{P}_p(d)$. Under this assumption, the estimated depth $\hat{d}(p)$ is usually defined as the expectation of this distribution, approximated as the integral of the product of depth hypothesis $\{d_m\}_{m=1}^{M}$ and its estimated probability along the ray: $\hat{d}(p) = \mathbb{E}[\mathbf{P}_p(d)] \approx \sum_{d \in \{d_m\}_{m=1}^{M}} d\mathbf{P}_p(d)$.

The unimodal depth distribution is a valid assumption if the discrete depth map is of sufficiently high resolution and can approximate the continuous depth distribution well [17]. However, pixels at lower resolutions could be the projection of a group of 3D points with different depth values, especially for 3D structures across the boundaries of an object with depth discontinuities, which, as shown in Fig. 3, are inherent of a multi-modal distribution.

As shown in Fig. 4a, existing cascaded cost volume-based works using a unimodal distribution to represent those pixels might result in incorrect depth estimation. The estimated depth, defined as the expectation of the distribution, might be too far away from any of the depth modes; therefore, it would not be possible to recover it during the following refinement steps. That is, the algorithm makes an inaccurate early decision, the error of which will be propagated to subsequent modules. Instead, we introduce a non-parametric depth probability modeling for each pixel to deal with pixels with arbitrary distributions. Specifically, given a pixel p at the coarse level l, its depth d_p follows a continuous probability distribution. We approximate this continuous distribution $\mathbf{P}^l(d_p)$ on a set of discrete depth hypotheses (discrete samples) $\{d_{p,m}^l\}_{m=1}^{M^l}$.

Next, we introduce our framework for depth inference based on this non-parametric depth probability distribution modeling.

3.2. Cost volume pyramid

We construct the cost volume pyramid for depth inference using a feature pyramid to extract features $\mathbf{f}^l$, $l \in \{0...L\}$, where $l = L$ refers to the coarsest level in the smallest resolution and $l = 0$ refers to the finest level corresponding to the full resolution.

Regular cost volume for depth initialization. We first build the cost volume at level L. Assume $\{\mathbf{f}_i^L\}_{i=0}^{N}$ refers to the feature maps extracted from the reference view and the N source views.

Given a pre-defined global depth searching range $[d_{min}, d_{max}]$, we uniformly sample $\{d_m\}_{m=1}^{M^L}$ depth values corresponding to M^L fronto-parallel planes on the *inverse* depth space [20]. Here d_{min} and d_{max} represent the lower and upper bound of the searching range, respectively. Note that a sampled depth $d \in \{d_m\}_{m=1}^{M^L}$ represents a plane parallel to the image plane of the reference camera. We use the differential homography [23] computed for the plane defined by the depth d to warp features of the source view i to the reference view $\mathbf{f}_{i\to0,d}^L$. We compute the matching cost as the Group-wise correlation of G^L groups between the reference feature $\mathbf{f}_0^L$ and warped source feature $\mathbf{f}_{i\to0,d}^L$. Then, we estimate the cost maps for each depth hypothesis plane and concatenate them into a cost volume $C^L \in \mathbb{R}^{H^L \times W^L \times M^L \times G^L}$. Here W^L, and H^L denote the image size at level L. We adopt a view aggregation module similar to [20] to estimate a visibility map and fuse matching cost computed from different source views.

Given this regular cost volume C^L, we use a regular 3D-UNet similar to [22] for cost aggregation. The output of this initial cost aggregation network is a probability volume denoted as $P^L \in \mathbb{R}^{H^L \times W^L \times M^L}$ which defines the non-parametric depth probability distribution for each pixel, represented by the probability for each depth sample. Then, we explore the pixel-wise depth samples with the top K probabilities to build the cost volume for the next level.

Sparse Cost volume for depth refinement. Without loss of generality we ignore the pixel index from now on. Let $\{d_{Q_i}^l\}_{i=1}^{K}$ define the depth samples of top K estimated probability at level l, where $\{Q_i\}_{i=1}^{K}$ define the top K indices and Δd^l is the corresponding depth search interval. To perform local search around the K possible depth samples obtained from level l, we define the depth samples for level $l - 1$ by subdividing each selected depth sample at level l into two samples, see Fig. 5. This process is formulated for each
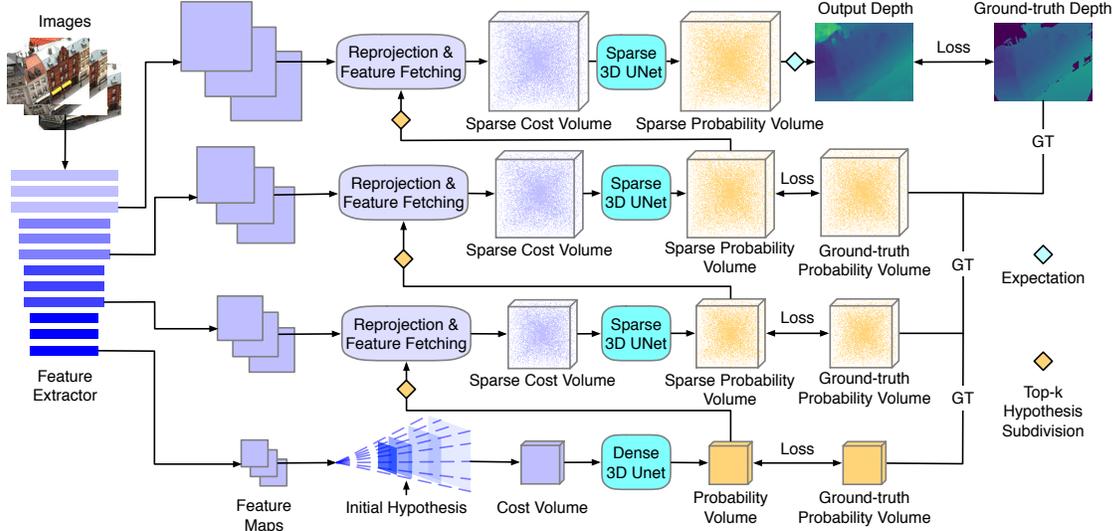
Figure 2. Network Structure. We first build the feature pyramid from the source and reference images. We then build the cost volume pyramid based on the modelling of pixel-wise non-parametric depth probability distribution. Specifically, the cost volume at each level is built based on the depth samples of top K probability from the previous level. The cost volume is sparse and aggregated by sparse convolution. The depth map $\mathbf{D}^0$ is inferred at the full resolution level.
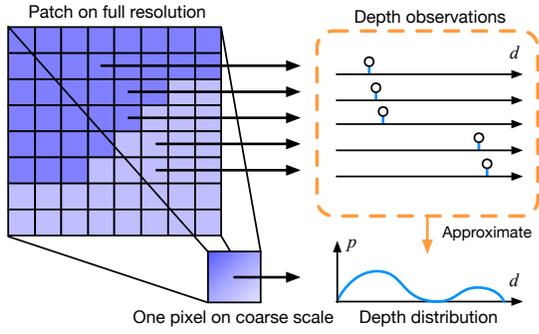

Figure 3. The depth distribution of a coarse pixel can be approximated by the depth observations in the corresponding patch on full resolution depth map

pixel as

$$d_{m-}^l = d_m^l - \frac{1}{4}\Delta d^l$$
$$d_{m+}^l = d_m^l + \frac{1}{4}\Delta d^l, \tag{1}$$

where $m \in \{Q_i\}_{i=1}^K$. The depth samples for level $l-1$ are estimated as $\mathbf{S}^{l-1} = \{d_{m-}^l, d_{m+}^l | m \in \{Q_i\}_{i=1}^K\}$ with the updated depth searching interval $\Delta d^{l-1} = 0.5\Delta d^l$.

Due to the resolution difference across levels, $\mathbf{S}^{l-1}$ is shared by pixels within a patch at level $l-1$ that corresponds to a pixel at level l. Given the pixel-wise depth samples $\mathbf{S}^{l-1}$, we then build the cost volume to model the depth probability distributions for pixels at level $l-1$. As depth samples are formed in a pixel-wise manner, the relative spatial locations among neighboring 3D points are not reserved. We therefore leverage the sparse cost volume structure and aggregate information based on sparse convolutions. To this end, we define the sparse cost volume

as $C^{l-1} = \{(p_k, c_k)\}_{k=1}^{K^{l-1}}$, where c_k defines the matching cost, $p_k = (x_k, y_k, z_k)$ defines the 3D coordinate computed from a depth sample, K^{l-1} defines the total number of 3D points converted from depth samples. Given the camera projection matrix, we project p_k to other source views and the cost is similarly calculated as that for pixels at level L as described in the previous section.

Sparse cost aggregation network. As the sparse cost volume can not be efficiently aggregated by regular dense 3D convolutions, we build a sparse cost aggregation network to aggregate cost utilizing the rigid spatial relation stored in p_k. Specifically, the basic block of our network consists of three layers of sparse 3D convolution with factorized kernels on each dimension, a sparse batch normalization layer and a sparse ReLU activation [15]. We provide the detailed structure of the network in the supplementary material. The output of the sparse cost aggregation network is the probability distribution $\mathbf{P}_p^{l-1}(d)$, which is used as input to build the cost volume in the next refinement level, see Fig. 6 (b).

3.3. Depth Inference at full resolution

Unlike existing cascade cost volume-based approaches, we only perform the depth inference at the full resolution level 0. At this level, we approximate the depth of each pixel as the expectation of the estimated distribution,

$$\hat{d}(p) = \mathbb{E}(\mathbf{P}_p^0(d)) = \sum_{m=1}^{M^0} d_m^0 \mathbf{P}_p^0(d_m^0). \tag{2}$$
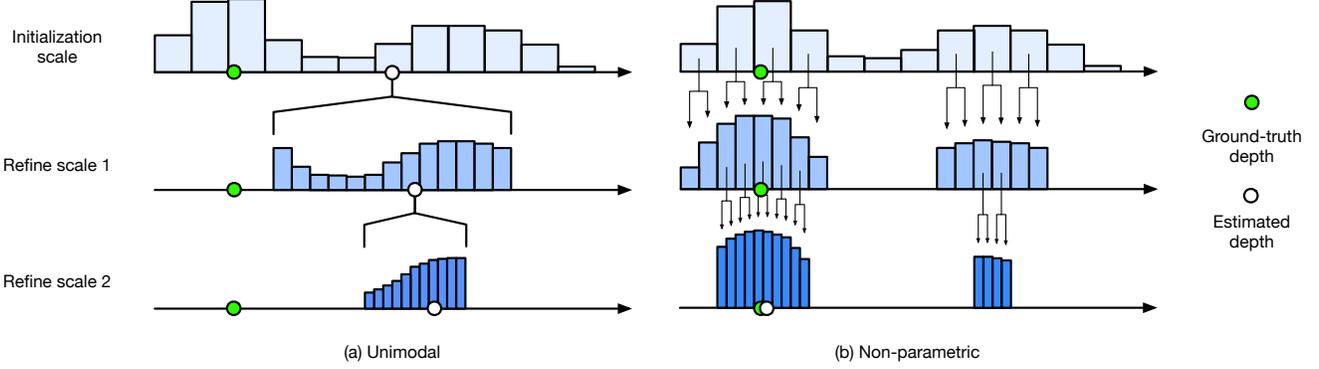
Figure 4. Unimodal and non-parametric depth searching. (a) Existing method assume unimodal distribution that can cause incorrect depth estimation. (b) Our non-parametric depth modeling can estimate correct depth from multi-modal depth distribution.
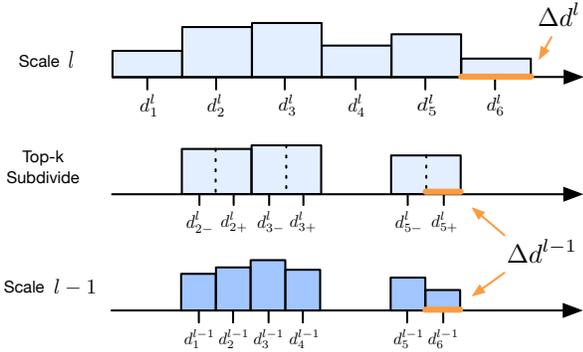

Figure 5. We generate new hypothesis by selecting top-k hypothesis from coarse level and equally subdivide them along the ray.

3.4. Loss Function

We train the network in a supervised manner. We use the depth probability distribution approximated by the high-resolution depth map observations as ground-truth. In particular, for each pixel, p, the ground-truth probability distribution $\mathbf{P}^l_{gt,p}$ is approximated by the histogram of depth observations from the corresponding patch Φ_p at the full-resolution, normalized by the sum of observations,

$$\mathbf{P}^l_{gt,p}(d^l_m) = \frac{Hist^l_p(d^l_m)}{\sum_{m=1}^{m=M^l} Hist^l_p(d^l_m)},$$

$$Hist^l_p(d^l_m) = \sum_{p' \in \Phi_p} \begin{cases} 1 - \frac{|d_{p'} - d^l_m|}{\Delta d^l}, & \text{if } |d_{p'} - d^l_m| \leq \Delta d \\ 0, & \text{if } |d_{p'} - d^l_m| > \Delta d, \end{cases}$$
(3)

where $d_{p'}$ is the ground-truth depth value of pixel p' in patch Φ_p, d^l_m is the depth hypothesis and Δd^l is the interval between depth hypotheses.

Finally, for each hypothesis d^l_m of pixel p, we compute the loss as the binary cross entropy between the estimated probability $\mathbf{P}^l_p(d^l_m)$ and ground-truth probability $\mathbf{P}^l_{gt,p}(d^l_m)$,

$$\mathcal{L}^l_p(d^l_m) = BCE(\mathbf{P}^l_p(d^l_m), \mathbf{P}^l_{gt,p}(d^l_m)).$$
(4)

Empirically, we observe that the ground-truth probability distribution usually concentrates on a few hypotheses, causing an imbalanced number of samples with zero and with non-zero probability. We address this problem balancing the loss as,

$$\mathcal{L}^l = \sum_{p \in \Omega^l} \sum_{m=1}^{M^l} \lambda^l_p(d^l_m) \mathcal{L}^l_p(d^l_m),$$
(5)

$$\lambda^l_p(d^l_m) = \begin{cases} 1 - \sigma^l, & \text{if } \mathbf{P}^l_{gt,p}(d^l_m) > 0 \\ \sigma^l, & \text{if } \mathbf{P}^l_{gt,p}(d^l_m) = 0 \end{cases}$$
(6)

$$\sigma^l = \frac{\sum_{m=1}^{M^l} \sum_{p \in \Omega^l} \mathbb{1}(\mathbf{P}^l_{gt,p}(d^l_m) > 0)}{H^l \times W^l \times M^l},$$
(7)

where Ω^l defines the image coordinate domain at level l, σ^l represents the percentage of hypothesis with probability larger than zero. At the final level, we supervise the depth estimation by the l_1 norm measuring the distance between ground-truth depth map and final estimated depth map.

$$\mathcal{L}^0 = \sum_{p \in \Omega^0} ||\mathbf{D}^0_{gt}(p) - \mathbf{D}^0(p)||_1.$$
(8)

The overall loss is a weighted sum of the BCE loss on coarse scales and the l_1 loss on the final level,

$$\mathcal{L} = \sum_{l=0}^{L} w^l \mathcal{L}^l$$
(9)

where w^l is the weight for the loss at level l.

4. Experiments

In this section, we demonstrate the performance of our approach with a comprehensive set of experiments in standard benchmarks. Below, we first describe the datasets and benchmarks, the implementation details, and then, present and analyze our results.
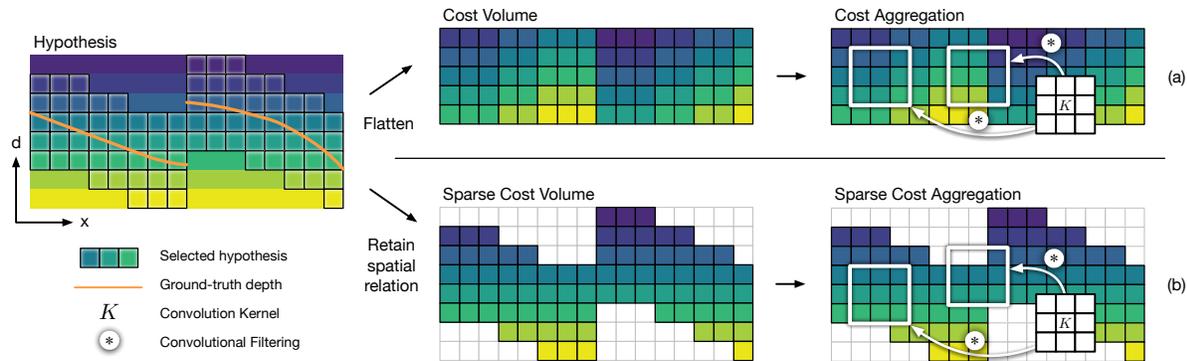
Figure 6. Sparse cost volume and sparse cost aggregation. Color indicates relative depth of hypothesis. (a) Existing methods build flattened cost volume that have spatial ambiguity. (b) We build sparse cost volume and use sparse cost aggregation to retain rigid spatial relation.

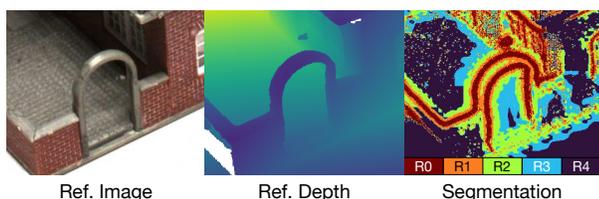Ref. Image   Ref. Depth   Segmentation

Figure 7. We segment ground-truth depth map into five regions corresponding to different depth smoothness and evaluate depth estimation accuracy on each region.

4.1. Datasets

DTU dataset [1] is a large-scale MVS dataset providing 124 scenes captured from 64 views under 7 lighting conditions. 3D points captured by structured-light scanners are provided for each scene as reference reconstruction. The dataset provides a color image and the corresponding camera parameters for each view.

BlendedMVS dataset [25] is a large-scale synthetic MVS dataset. It contains more than 17000 MVS training samples covering a variety of 113 scenes, including architectures, sculptures and small objects.

Tanks and Temples [6] dataset provides both indoor and outdoor scenes under real-life lightning conditions with large scale variations. We evaluate the generalization ability of our model on this public benchmark dataset.

4.2. Implementation

We implement the proposed model using *pytorch* [10] and use the *Torchsparse* [16] library for the sparse cost volume and the sparse cost aggregation.

Training We train our model on the DTU dataset using downsampled and cropped images of size 640×512 and their corresponding depth map. This map is rendered from a surface mesh generated by Screened Poisson Surface Reconstruction of the provided reference point cloud. We use the same split of training, validation and test set as defined in [23]. We set the number of levels to $(L + 1) = 4$ and the number of hypothesis to $\{M^l\}_{l=0}^{L} = \{8, 16, 32, 48\}$.

We use 5 views for training and adopt the similar anti-noise training strategy as in [18]. We use Adam optimizer with 0.0005 learning rate and a batchsize of 2 on a Nvidia RTX Titan GPU. To demonstrate the generalization ability of our approach, we train our model on BlendedMVS [25] training set and test on Tanks and Temples dataset without any fine-tuning.

Metrics We follow the evaluation protocol used by [22–24]. On the DTU dataset, we report the *accuracy*, *completeness* and *overall score* of the reconstructed point clouds. *Accuracy* measures the distance from the estimated point clouds to the ground truth in millimeters and *completeness* measures the distance from ground truth point clouds to the estimated ones [1]. The *overall score* is the average of accuracy and completeness [23]. On Tanks and Temples, we report the *f-score* for each scene and the average of all scenes.

We also evaluate the depth estimation accuracy specifically on depth boundaries. To this end, we use a Laplacian pyramid simulating a band-pass filter to segment the ground-truth depth map into non-overlapping regions of different depth smoothness, see Fig. 7. We first build a depth map pyramid $\{\mathbf{D}_{gt}^q\}_{q=0}^{Q}$ of $Q+1$ levels by repeatedly downsampling the ground-truth depth map $\mathbf{D}_{gt}^0$. We then build the Laplacian pyramid of depth by taking the difference between adjacent levels in $\{\mathbf{D}_{gt}^q\}_{q=0}^{Q}$, where the lower level depth is upsampled to match the size of the upper one.

Each level of the Laplacian pyramid contains depth structures present at a particular scale, which we use as a mask to segment the depth map into different smoothness regions. In particular, we use 5 levels to segment the depth map into five regions $\{R0, R1, R2, R3, R4\}$, see Fig. 7. The region $R0$ corresponds to the highest level of the Laplacian pyramid and contains pixels with the most abrupt depth changes usually caused by depth boundaries or small objects. The lower levels correspond to regions with intermediate and smooth depths. We report the average depth error for each of the five region as the average l_1 distance in millimeter between estimated depth and ground-truth depth.

Ref. Image   Ground-truth   Unimodal baseline   PatchmatchNet [18]   Ours
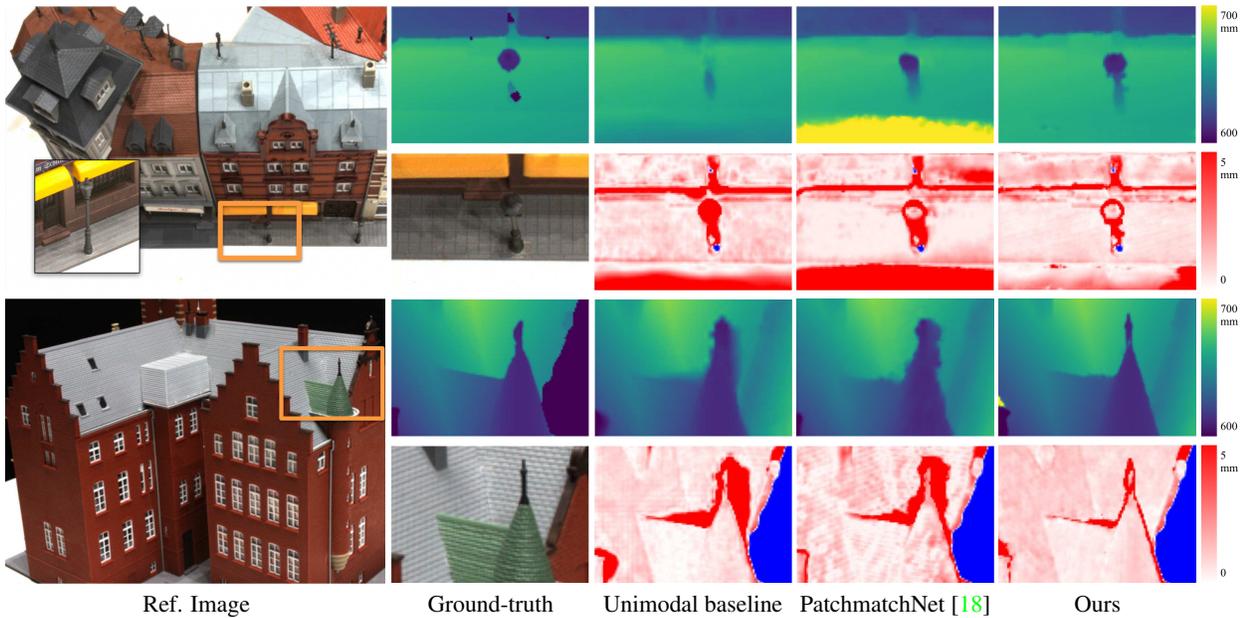
Figure 8. **DTU dataset.** Example qualitative results on small objects and depth boundaries. Every upper row shows the ground-truth and estimated depth maps and under row shows the estimation error comparing with ground-truth depth map. Areas with no ground-truth are marked blue. Our method based on non-parametric depth distribution modeling is more accurate on small objects and boundary regions.

Methods	Views	Acc.	Comp.	Overall
D2HC-RMVSNet [21]	7	0.395	0.378	0.386
Vis-MVSNet [26]	5	0.369	0.361	0.365
PVA-MVSNet [26]	7	0.379	0.336	0.357
AA-RMVSNet [19]	7	0.376	0.339	0.357
CasMVSNet [5]	5	0.325	0.385	0.355
EPP-MVSNet [8]	5	0.413	0.296	0.355
PatchmatchNet [18]	5	0.427	0.277	0.352
CVP-MVSNet [22]	5	**0.296**	0.406	0.351
UCSNet [3]	5	0.338	0.349	0.344
BP-MVSNet [14]	5	0.333	0.320	0.327
PVSNet [20]	11	0.337	0.315	0.326
Ours	5	0.356	**0.275**	**0.315**

Table 1. **DTU dataset.** Quantitative results of reconstruction quality; lower is better. Our model outperforms all existing methods on mean completeness and overall score.

Evaluation We set the number of hypothesis per level as $\{M^l\}_{l=0}^L = \{8, 16, 32, 96\}$ for testing. On the DTU dataset, we test on the full-resolution images of size 1600×1184 and set number of views to $N = 5$ with [4] for fusion. On the Tanks and Temples dataset, we use images of size 1920×1080 with $N = 11$ views, camera parameters generated by Colmap [11,12] and the fusion method in [21].

4.3. Results on DTU dataset

We first compare the overall reconstruction quality of our proposed method with existing methods on the DTU dataset. As summarized in Tab. 1, our method outperforms all existing methods on both *mean completeness* and *overall score*. We also analyze the quality of the reconstruction on the boundaries using the average depth error on the five

Method	R0	R1	R2	R3	R4
D2HC-RMVSNet [21]	7.20	2.33	2.59	5.14	10.2
PatchmatchNet [18]	9.54	2.37	2.40	4.72	9.15
Unimodal baseline	8.85	2.39	2.15	3.97	9.08
Ours	**6.48**	**1.94**	**1.81**	**3.85**	**8.61**

Table 2. **DTU dataset.** Performance on regions of different depth smoothness. Our method can achieve lowest error on boundary region (R0).

regions of the Laplacian pyramid. For comparison, we consider PatchmatchNet [18] and a recurrent method D2HC-RMVSNet [21]. As a baseline, we consider a variation of our method using a unimodal distribution. As summarized in Tab. 2, our method outperforms the others on the boundary regions. Qualitative results in Fig. 1 and Fig. 8 also demonstrate our method is more accurate on boundary regions and small objects producing sharper depth discontinuities. Efficiency-wise, our model takes 6054 MB GPU memory and 1.2s to estimate a full-resolution depth map, which is on par with existing cost volume-based methods.

4.4. Results on Tanks and Temples

In this experiment, we evaluate the generalization ability of the proposed method on the Tanks and Temples dataset [6]. The summary of quantitative results is listed in Tab. 3 and representative qualitative results are shown in Fig. 9. In these results, we can observe that our approach produces sharp and accurate depth estimation on boundary regions and, overall, our model trained on the BlendedMVS dataset achieves competitive performance compared to other recent methods in the benchmark.

Figure 9. **Tanks and Temples dataset.** Example point cloud results. The second row shows details of first row. Best viewed on screen.

Methods	Mean	Family	Francis	Horse	Lighthouse	M60	Panther	Playground	Train
AA-RMVSNet [19]	61.51	77.77	59.53	51.53	64.02	64.05	59.47	60.85	54.90
EPP-MVSNet [8]	61.68	77.86	60.54	52.96	62.33	61.69	60.34	62.44	55.30
Vis-MVSNet [26]	60.03	77.40	60.23	47.07	63.44	62.21	57.28	60.54	52.07
Ours	59.64	78.93	64.09	51.82	59.42	58.39	55.71	56.07	52.71
D2HC-RMVSNet [21]	59.20	74.69	56.04	49.42	60.08	59.81	59.61	60.04	53.92
BP-MVSNet [14]	57.60	77.31	60.90	47.89	58.26	56.00	51.54	58.47	50.41
PVSNet [20]	56.88	74.00	55.17	39.85	61.37	60.22	56.87	58.02	49.51
PatchmatchNet [18]	53.15	66.99	52.64	43.24	54.87	52.87	49.54	54.21	50.81
PatchMatch-RL [7]	51.81	60.37	43.26	36.43	56.27	57.30	53.43	59.85	47.61

Table 3. **Tanks and Temples dataset.** Quantitative comparison to existing approaches. Our model can achieve competitive performance on Tanks and Temples dataset.

Method	Depth distribution		Cost aggregation		Acc.	Comp.	Overall
	Unimodal	Non-parametric	Standard	Sparse			
Baseline	✓		✓		0.3498	0.3351	0.3425
Baseline + non-parametric modeling		✓	✓		0.3387	0.3792	0.3590
Baseline + sparse cost aggregation	✓			✓	0.3538	0.3096	0.3317
Ours		✓		✓	0.3563	0.2750	**0.3156**

Table 4. **DTU dataset.** Quantitative comparison of the contribution of each module in our framework. Using non-parametric depth distribution modeling with standard 3D convolutions can cause performance drop due to spatial ambiguity. Our proposed sparse cost aggregation can improve the performance of unimodal based methods. The proposed non-parametric depth distribution modeling requires the sparse cost aggregation to achieve best reconstruction quality.

4.5. Ablation Study

We provide ablation experiments on the DTU dataset to evaluate the contribution of the proposed modules. We start with a baseline model using standard unimodal distribution and standard 3D convolutions for cost aggregation. Results are shown in Tab. 4. Using non-parametric depth distribution modeling with standard 3D convolutions can cause performance drop due to spatial ambiguity. Our proposed sparse cost aggregation can improve the performance of unimodal based methods. The proposed non-parametric depth distribution modeling requires the sparse cost aggregation to achieve best reconstruction quality.

Limitations. Our approach leverages sparse cost volume and sparse convolutions to yield better accuracy. However, sparse convolutions are computationally expensive as they are not fully optimized. This increases the inference time.

We plan to improve the efficiency in our future work.

5. Conclusion

We proposed an approach for depth inference based on modelling non-parametric depth probability distribution for each pixel. Our modelling can handle pixels with unimodal and multi-modal depth distributions such as pixels on the boundaries. Our method does not infer depth at the coarse level which avoids the depth errors made at the early stage and following propagation to the refinement levels. Experimental results show that our approach can achieve superior performance especially for pixels on the boundaries.

Acknowledgments

References

[1] Henrik Aanæs, Rasmus Ramsbøl Jensen, George Vogiatzis, Engin Tola, and Anders Bjorholm Dahl. Large-scale data for multiple-view stereopsis. *IJCV*, 2016. 6

[2] Rui Chen, Songfang Han, Jing Xu, and Hao Su. Point-based multi-view stereo network. In *ICCV*, 2019. 2

[3] Shuo Cheng, Zexiang Xu, Shilin Zhu, Zhuwen Li, Li Erran Li, Ravi Ramamoorthi, and Hao Su. Deep stereo using adaptive thin volume representation with uncertainty awareness. In *CVPR*, 2020. 1, 2, 7

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

[5] Xiaodong Gu, Zhiwen Fan, Siyu Zhu, Zuozhuo Dai, Feitong Tan, and Ping Tan. Cascade cost volume for high-resolution multi-view stereo and stereo matching. In *CVPR*, 2020. 1, 2, 7

[6] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics (ToG)*, 2017. 6, 7

[7] Jae Yong Lee, Joseph DeGol, Chuhang Zou, and Derek Hoiem. Patchmatch-rl: Deep mvs with pixelwise depth, normal, and visibility. In *CVPR*, October 2021. 8

[8] Xinjun Ma, Yue Gong, Qirui Wang, Jingwei Huang, Lei Chen, and Fan Yu. Epp-mvsnet: Epipolar-assembling based depth prediction for multi-view stereo. In *ICCV*, pages 5732–5740, October 2021. 7, 8

[9] Yamin Mao, Zhihua Liu, Weiming Li, Yuchao Dai, Qiang Wang, Yun-Tae Kim, and Hong-Seok Lee. Uasnet: Uncertainty adaptive sampling network for deep stereo matching. In *CVPR*, pages 6311–6319, 2021. 2

[10] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *NeurIPS*, pages 8024–8035. Curran Associates, Inc., 2019. 6

[11] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *CVPR*, 2016. 7

[12] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *ECCV*, 2016. 7

[13] Steven M Seitz, Brian Curless, James Diebel, Daniel Scharstein, and Richard Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *CVPR*, 2006. 1

[14] Christian Sormann, Patrick Knöbelreiter, Andreas Kuhn, Mattia Rossi, Thomas Pock, and Friedrich Fraundorfer. Bp-mvsnet: Belief-propagation-layers for multi-view-stereo. In *3DV*, pages 394–403. IEEE, 2020. 7, 8

[15] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *CVPR*, pages 2818–2826, 2016. 4

[16] Haotian* Tang, Zhijian* Liu, Shengyu Zhao, Yujun Lin, Ji Lin, Hanrui Wang, and Song Han. Searching efficient 3d architectures with sparse point-voxel convolution. In *ECCV*, 2020. 6

[17] Fabio Tosi, Yiyi Liao, Carolin Schmitt, and Andreas Geiger. Smd-nets: Stereo mixture density networks. In *CVPR*, pages 8942–8952, 2021. 2, 3

[18] Fangjinhua Wang, Silvano Galliani, Christoph Vogel, Pablo Speciale, and Marc Pollefeys. Patchmatchnet: Learned multi-view patchmatch stereo, 2021. 1, 2, 6, 7, 8

[19] Zizhuang Wei, Qingtian Zhu, Chen Min, Yisong Chen, and Guoping Wang. Aa-rmvsnet: Adaptive aggregation recurrent multi-view stereo network. In *ICCV*, pages 6187–6196, October 2021. 2, 7, 8

[20] Qingshan Xu and Wenbing Tao. Pvsnet: Pixelwise visibility-aware multi-view stereo network. *arXiv*, 2020. 2, 3, 7, 8

[21] Jianfeng Yan, Zizhuang Wei, Hongwei Yi, Mingyu Ding, Runze Zhang, Yisong Chen, Guoping Wang, and Yu-Wing Tai. Dense hybrid recurrent multi-view stereo net with dynamic consistency checking. In *ECCV*, 2020. 2, 7, 8

[22] Jiayu Yang, Wei Mao, Jose M Alvarez, and Miaomiao Liu. Cost volume pyramid based depth inference for multi-view stereo. In *CVPR*, 2020. 1, 2, 3, 6, 7

[23] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. In *ECCV*, 2018. 1, 2, 3, 6

[24] Yao Yao, Zixin Luo, Shiwei Li, Tianwei Shen, Tian Fang, and Long Quan. Recurrent mvsnet for high-resolution multi-view stereo depth inference. In *CVPR*, 2019. 1, 2, 6

[25] Yao Yao, Zixin Luo, Shiwei Li, Jingyang Zhang, Yufan Ren, Lei Zhou, Tian Fang, and Long Quan. Blendedmvs: A large-scale dataset for generalized multi-view stereo networks. *CVPR*, 2020. 6

[26] Jingyang Zhang, Yao Yao, Shiwei Li, Zixin Luo, and Tian Fang. Visibility-aware multi-view stereo network. *BMVC*, 2020. 7, 8