Exploiting Correspondences with All-pairs Correlations for Multi-view Depth Estimation

Kai Cheng, Hao Chen, Wei Yin, Guangkai Xu, Xuejin Chen

Abstract—Multi-view depth estimation plays a critical role in reconstructing and understanding the 3D world. Recent learning-based methods have made significant progress in it. However, multi-view depth estimation is fundamentally a correspondence-based optimization problem, but previous learning-based methods mainly rely on predefined depth hypotheses to build correspondence as the cost volume and implicitly regularize it to fit depth prediction, deviating from the essence of iterative optimization based on stereo correspondence. Thus, they suffer unsatisfactory precision and generalization capability. In this paper, we are the first to explore more general image correlations to establish correspondences dynamically for depth estimation. We design a novel iterative multi-view depth estimation framework mimicking the optimization process, which consists of 1) a correlation volume construction module that models the pixel similarity between a reference image and source images as all-to-all correlations; 2) a flow-based depth initialization module that estimates the depth from the 2D optical flow; 3) a novel correlation-guided depth refinement module that reprojects points in different views to effectively fetch relevant correlations for further fusion and integrate the fused correlation for iterative depth update. Without predefined depth hypotheses, the fused correlations establish multi-view correspondence in an efficient way and guide the depth refinement heuristically. We conduct sufficient experiments on ScanNet, DeMoN, ETH3D, and 7Scenes to demonstrate the superiority of our method on multi-view depth estimation and its best generalization ability.

Index Terms—Multi-view Depth estimation, Correlation, Correspondence, Iterative refinement

I. INTRODUCTION

MULTI-VIEW depth estimation is a fundamental problem in 3D vision. It aims to recover the pixel depths with known camera parameters and provides geometric information for many vision tasks, such as 3D reconstruction [1], 3D detection [2], and semantic segmentation [3]. Reviewing previous methods of multi-view depth estimation, it is crucial to establish point correspondences among different views. Traditional methods [4]–[7] use handcrafted similarity metrics, e.g., the sum of absolute differences and normalized cross-correlation, to describe the point correspondence between image pairs. However, those handcrafted local features are sensitive to illumination variation, low texture, and repetitive patterns. In recent years, many learning-based methods [8]–[15] have emerged for multi-view depth estimation. They make a promising improvement in depth estimation due to more robust correspondences based on features extracted by deep neural networks (DNNs) [6], [7]. These methods mainly follow the paradigm of cost volume construction and regularization for depth estimation, as Fig. 1 (a) shows. However, this pipeline still has shortcomings in the way of establishing and utilizing the correspondence. First, the pixel correspondence is predefined by the discrete depth hypotheses. Thus, the predicted depth is sensitive to the sampling strategy of the hypotheses and needs to be modified for different types of scenes. Second, the established correspondence only considers the sparse and fixed pixels from projection and ignores the relation from neighboring areas, which may lead to false matching in textureless or repeated pattern areas. Third, the cost volume which represents the correspondence is processed implicitly by DNNs to fit the depth prediction directly. It is more inclined to overfit the training data compared to the traditional iterative optimization based on dynamically established correspondence.

In essence, multi-view depth estimation is an optimization problem minimizing the pixel matching cost based on stereo correspondence. Traditional methods [16] have built an interpretable optimization pipeline, while the powerful representation ability of deep learning can make up for the deficiency of handcrafted correspondence expression. Therefore, we follow the traditional optimization-based framework but apply DNNs for image feature extraction and iterative optimization. Based on the distinctive features extracted by DNNs, dense and reliable correspondences can be established. With the help of the approximation ability of DNNs, depth can be better optimized by combining correspondence cue...
and other regularization terms such as context information. Moreover, the idea of utilizing deep learning to facilitate the correspondence representation and optimization has been verified in optical flow estimation [17] and shows superior performance in various scenarios compared to other methods.

Intuitively, considering multi-view depth estimation as a correspondence-based optimization problem, we propose a novel framework that iteratively optimizes the depth prediction based on projected correlations, as Fig. 1(b) shows. We can evaluate the quality of a depth prediction from the correlation volume by reprojecting the point into multiple views according to its estimated depth and checking the correlation between corresponding points. The retrieved correlations in turn facilitate the incremental depth refinement. Specifically, our framework is composed of three modules: 1) The correlation construction module decouples the pixel correspondences from predefined depth hypotheses and describes all-to-all pixel similarity in an efficient way. It helps to alleviate mismatching on confusing areas by provide neighboring dense correspondences for fusion in the refinement module. 2) In order to initialize the reasonable depth for further optimization and ensure its generalization ability, the depth initialization module leverages the correlation volumes to estimate the optical flow for predicting an initial depth by triangulation. 3) Correlation-guided depth refinement module mimics the steps of traditional optimization to iteratively update the depth prediction according to the correlations. We design a correlation fusion module and depth updating module to take full use of multi-view correlations in each iteration. The fusion of correlations is realized by explicitly projecting each point into multiple views according to its predicted depth. The depth updating module predicts the depth residual according to the fused correlation features, which describe the quality of the correspondence from the estimated depth. Additionally, to handle the problem of occlusion or non-overlapping, which cannot be solved by correspondence, we introduce a learnable context feature as a regularization term to the updating module. Taking efficiency into account, the depth is initialized and iteratively updated at a low resolution, and finally upsampled to the original resolution. The retrieved correlations in turn facilitate the incremental depth refinement. Specifically, our framework is composed of three modules: 1) The correlation construction module decouples the pixel correspondences from predefined depth hypotheses and describes all-to-all pixel similarity in an efficient way. It helps to alleviate mismatching on confusing areas by provide neighboring dense correspondences for fusion in the refinement module. 2) In order to initialize the reasonable depth for further optimization and ensure its generalization ability, the depth initialization module leverages the correlation volumes to estimate the optical flow for predicting an initial depth by triangulation. 3) Correlation-guided depth refinement module mimics the steps of traditional optimization to iteratively update the depth prediction according to the correlations. We design a correlation fusion module and depth updating module to take full use of multi-view correlations in each iteration. The fusion of correlations is realized by explicitly projecting each point into multiple views according to its predicted depth. The depth updating module predicts the depth residual according to the fused correlation features, which describe the quality of the correspondence from the estimated depth. Additionally, to handle the problem of occlusion or non-overlapping, which cannot be solved by correspondence, we introduce a learnable context feature as a regularization term to the updating module. Taking efficiency into account, the depth is initialized and iteratively updated at a low resolution, and finally upsampled to the original resolution in a coarse-to-fine manner to recover details.

In summary, we make the following contributions for multi-view depth estimation in this paper:

- Distinct from previous deep multi-view depth estimation methods that rely on predefined depth hypotheses to build correspondence as cost volume and implicitly predict the depth from cost volume regularization, we are the first to explore pixel correlations to build a novel framework that facilitates multi-view depth estimation.
- We design a correlation-guided depth refinement module to exploit correspondences for depth prediction iteratively. An effective correlation fusion strategy is proposed to dynamically assess the quality of the current correspondences from projection and guide depth updating.
- Our correlation-based multi-view depth estimation framework is effective and generalized to various scenarios. It produces more precise depth prediction as well as higher visual quality. We demonstrate our new state-of-the-art performance on ScanNet [18], DeMoN [19], and the best generalization ability on ETH3D [20] and 7Scenes [21].

II. RELATED WORK

Multi-view depth estimation is a popular solution for multi-view stereo (MVS), which focuses on recovering the 3D geometry of the scene from the pixel correspondences among the views. In this section, we will revisit the traditional MVS methods first and then summarize the current learning-based multi-view depth estimation methods. Since we design a novel multi-view depth estimation framework from the perspective of the representation and usage of correspondence, we also introduce the research of correspondence in the field of depth estimation and 3D reconstruction.

Traditional Multi-view Stereo. Traditional MVS approaches can be summarized into three categories, including point-based [22], [23], voxel-based [24]–[27], and depth-based [1], [4], [28]–[32]. The point-based and voxel-based methods directly make predictions in 3D space with high consumption of computation and memory. In contrast, depth-based methods are more flexible and efficient by describing the 3D structure through a single scalar for each point. They establish the correspondence between images from multiple views by plane sweep [5] or patch match [33] and then estimate the depth values by minimizing a global matching cost. Furthermore, the estimated depth map can be converted to implicit fields for reconstructing point clouds [24] or 3D voxels [35]. In this paper, we further explore the depth-based MVS by cooperating DNNs. The focus of our work is per-view depth estimation. The final reconstruction can be achieved by integrating any depth fusion methods [36], [57].

Learning-based Multi-view Depth Estimation. In recent years, learning-based multi-view depth estimation [8]–[15], [58]–[45] have made great progress. DeepMVS [8] builds a set of cost volumes by plane sweep [5] from an arbitrary number of image patches and then aggregates these volumes with a ConvNet to infer depth maps. However, the global information is lost for DeepMVS because it pre-processes the image as patches. For better learning the global information, MVSNet [10] and DPSNet [11] directly extract the features from the whole image and construct the cost volume in the feature space. Their common idea is to build a matching cost volume among views based on a set of depth hypotheses and then regularize the cost volume by 3D convolutions to predict the depth map.

To further improve the performance, R-MVSNet [41] reduces the memory consumption by replacing 3D convolution with gated recurrent units (GRU) to sequentially regularize the 2D cost maps along the depth direction. NAS [12] improves depth estimation by joint learning of normal maps. DeepVideoMVS [14] and EST [15] extend MVS to videos by fusing temporal information based on long short-term memory or Transformer. However, constructing a cost volume requires a set of predefined discrete depth hypotheses, which inevitably harms the precision and generalization ability. In contrast, we directly make depth predictions based on multi-view pixel-wise correlations rather than predefined depth hypotheses.

An exception of this cost-volume-based framework is DELTAS [42]. To avoid the high memory consumption in cost
volume regularization, it first predicts the depths for sparse keypoints by triangulation and then densifies the depth map using 2D convolutions. Inspired by their two-step framework, we propose an iterative depth refinement framework and build dense point-wise correspondences instead of sparse correspondences to fully exploit the multi-view correlations.

**Image Correspondence for 3D Reconstruction.** Establishing point correspondences among multi-view images is a fundamental step for 3D reconstruction. It is common to build sparse correspondences in structure from motion (SfM) [46], [47] and visual simultaneous localization and mapping (SLAM) [48], [49]. The feature point detectors and descriptors can be manually designed [50], [51] or trained [52], [53] to establish the sparse correspondences. Besides, the optical flow can also be converted as pixel correspondences to provide photometric restriction for SLAM. In comparison, MVS systems typically build dense correspondences based on handcrafted features or learned features by DNNs. Cost volume and correlation volume are the two popular representations of correspondences. Cost volumes represent the pixel correspondences by plane sweep based on a set of depth hypotheses. In comparison, correlation volumes directly depict the pixel-wise similarity between two images and are widely used for stereo matching [54], [55] and optical flow estimation [17], [56]. In this paper, we employ correlation volumes to depict pixel correspondences and fully exploit the dense point correlations for accurate depth estimation.

### III. Our Method

#### A. Overview

Given a series of images including one image \( I \) under a reference view and other source images \( \{ I_k \}_{k=1}^m \) with its camera intrinsic matrix \( K \) and camera poses \( \{ T, T_1, \ldots, T_m \} \), the goal of multi-view depth estimation is to predict a depth map \( D \) under the reference view. Instead of building cost volumes from a set of depth hypotheses, we build correlation volumes between images for iterative depth estimation. Fig. 2 shows the framework of our method, which consists of three parts:

(a) **Correlation volume construction.** To depict the dense point-wise correspondences between the reference view and other source views, we build a pyramid of correlation volumes based on the encoded features. The correlation pyramid avoids predefined depth hypotheses and represents multi-scale dense correspondence with neighboring relationships to alleviate mismatching in confusing areas.

(b) **Flow-based depth initialization.** Based on the correlation pyramid, we estimate optical flows between the reference view and other views to establish dense point correspondences. Then we predict an initial depth map by multi-view triangulation according to the estimated dense correspondences. This module can be generalized to various scenes and provides a reasonable initial point for subsequent refinement.

(c) **Correlation-guided depth refinement.** After initialization, a recurrent unit consisting of correlation fusion and depth updating is designed to refine the depth values progressively considering the point correlations based on the depth values predicted at the last iteration. This design dynamically establishes the multi-view correspondence to reflect the quality of the currently estimated depth and heuristically guides depth updating. Considering efficiency, the entire process is carried out in low resolution and finally upsampled in a coarse-to-fine manner.

#### B. Correlation Volume Construction

We extract the correlation volume pyramid to depict the pixelwise similarities between the reference view and other source views. For the reference image \( I \) and a source image \( I_k \), their image feature maps \( E = f_0(I) \) and \( E_k = f_0(I_k) \) are first extracted by a feature encoder \( f_0 \). Denote \( H \) and \( W \) as the height and width of the feature maps, which are one-eighth of the image resolution. A 4D correlation volume \( C(I, I_k) \in \mathbb{R}^{H \times W \times H \times W} \) is calculated as:

\[
C(I, I_k)_{p,q,r,s} = \sum_{u,v} E_{p+u,q+v} \cdot E_{k,r+u,k,s+v}
\]

where \( (p,q) \) and \( (r,s) \) are the pixel coordinates in the feature maps of \( I \) and \( I_k \), respectively. The correlation volume is then upsampled to the original resolution and used for depth estimation.
The first correlation volume on the last two dimensions. For collecting sufficient neighboring information, we construct the local neighborhood of \( p \) \( R \) the feature vectors of a point \( p \) \( I \) source images:

\[
\text{minimize the projection errors from the reference view to all}
\]

computes a depth \( d \) to represent the correspondences. Then we compute an initial depth prediction is inevitably flawed because of insufficient resolution for distant points. Our key idea is to extract the correlation vector \( v^l \) for \( p \), its corresponding pixel \( p_k \) in source image \( I_k \) is first projected by the current depth \( d_{t-1} \). Then \( p \) and the local neighborhood of \( p_k \) are used to index the correlations in \( C^l \) as \( v^l \).

\[
\mathbb{R}^{H \times W \times H \times W} \text{ is calculated by taking the dot product between}
\]

the feature vectors of a point \( p \) \( \begin{pmatrix} x, y \end{pmatrix}^T \) in the reference image and a point \( q \) \( \begin{pmatrix} u, v \end{pmatrix}^T \) in the source image as

\[
C^l(p, q) = E(p) \cdot E_k(q). \quad (1)
\]

In order to represent the point correlations at different scales for collecting sufficient neighboring information, we construct a 4-layer correlation pyramid \( \{ C^l \mid l = 0, \ldots, 3 \} \) by pooling the first correlation volume on the last two dimensions. For each layer, the correlation volume \( C^l \in \mathbb{R}^{H \times W \times 2^l \times 2^l} \).

\[ \text{C}^{l}(p, q) = \mathbf{E}(p) \cdot \mathbf{E}_k(q). \] (1)

C. Flow-based Depth Initialization

According to multi-view geometry, the depth of a point can be calculated by triangulation based on multi-view point correspondences. We first estimate the optical flow \( O_k \) between the reference image and each source image using RAFT [17] to represent the correspondences. Then we compute an initial depth for each pixel by multi-view triangulation as follows.

With the known correspondence and relative pose \( \{ \mathbf{R}_k, t_k \} \) of the reference image to the source images, the triangulation computes a depth \( d \) for a pixel \( p \) in the reference image to minimize the projection errors from the reference view to all source images:

\[
E_{proj} = \sum_k \left( \left( (K^{-1} \mathbf{p}_k) \times (\mathbf{R}_k K^{-1} \mathbf{p} + t_k) \right) \right)^2. \tag{2}
\]

The relative pose \( \mathbf{R}_k t_k \) can be derived from camera poses \( T \) and \( T_k \). \( \mathbf{p} = (x, y, 1)^T \) is the homogeneous coordinate for a pixel located at \( (x, y) \) in the reference image. \( \mathbf{p}_k \) is the homogeneous coordinate of its corresponding point \( (u_k, v_k) \) in the source image \( I_k \).

D. Correlation-Guided Depth Refinement

The initial depth prediction is inevitably flawed because of inaccurate optical flow estimation, pure camera rotation, and insufficient resolution for distant points. Our key idea is to iteratively revise the depth prediction based on the correlations extracted by the fusion module. The fused correlations describe the quality of the correspondences established from the estimated depth and guide the network to update the depth. Trading off the accuracy and efficiency, the depth map is initialized and iteratively refined at the low resolution and finally upsampled to the original resolution in a coarse-to-fine manner, as Fig. 2(c) shows.

1) Iterative Depth Updating: We first update the initial depth map iteratively at a low resolution, i.e., \( \mathbf{D}_0 \) in the resolution of \( H \times W \). In each iteration, we perform correlation fusion via reprojecting each point from the reference view to the source views according to the estimated depth in the last iteration and depth updating based on the fused correlations.

Correlation Fusion. At the \( t \)-th iteration, we have a depth map \( \mathbf{D}_{t-1} \) under the reference view estimated at the last iteration. For each pixel \( p \) in the reference image, we unproject it to the 3D space according to its current depth \( d_{t-1} \) and then reproject it to a source view as

\[
\mathbf{p}_k \sim K (\mathbf{R}_k K^{-1} d_{t-1} \mathbf{p} + t_k). \tag{3}
\]

As Fig. 3 shown, based on the reprojected position \( \mathbf{p}_k \), we fuse the correlations in its local neighborhood \( \mathcal{N}(\mathbf{p}_k) \) which contains all points \( \{ q \} \) that \( \| q - \mathbf{p}_k \|_1 \leq r \), where \( r \) is a constant. For each correlation volume \( C^l \) in the correlation pyramid, we take the correlation values \( C^l(p, q) \) for all points \( q \in \mathcal{N}(\mathbf{p}_k) \) and concatenate them together with their bilinear-interpolated value and form a correlation vector \( v^l \). A further concatenation of \( \{ v^l \} \) from all correlation volumes \( C^l \) forms a fused correlation feature for the point \( p \). With similar reprojection and correlation fusion for all the pixels in the reference image to a source image \( I_k \), we can obtain a correlation map \( \mathbf{V}_k \). We fuse the correlation maps \( \{ \mathbf{V}_k \}_{k=1}^m \) of the \( m \) source images into one correlation map \( \mathbf{V} \) to aggregate the point correspondences from multiple views. Our default fusion strategy \( \mathcal{F}(\cdot) \) is averaging as defined in Eq. 4. We also compare some other fusion strategies in Sec. IV-F.

\[
\mathbf{V} = \mathcal{F}(\mathbf{V}_1, \ldots, \mathbf{V}_m) = \frac{1}{m} \sum_{k=1}^m \mathbf{V}_k. \tag{4}
\]

Depth Updating. Based on the fused correlation map \( \mathbf{V} \) and depth map \( \mathbf{D}_{t-1} \) at the previous iteration, we update the depth from a convolutional GRU, as Fig. 2(c) shown. The correlation map \( \mathbf{V} \) and depth map \( \mathbf{D}_{t-1} \) are first separately passed through three \( \times 3 \times 3 \) convolutional layers and then concatenated together in the channel dimension as \( \mathbf{H} \) before being inputted into the GRU. However, the correlation map models multi-view correspondence, which can not totally solve the problem from non-overlapping or occlusion. As with traditional optimization methods adding some priors as regularization terms, we additionally integrate a learnable context feature \( \mathbf{F}^3 \) extracted from the reference image with \( \mathbf{H} \) to better handle the non-overlapping or occlusion regions. The details of context feature \( \mathbf{F}^3 \) are described in the following depth upsampling section. The specific architecture of GRU is the same as the one used in RAFT [17]. A residual depth map \( \Delta \mathbf{D}_t \) is predicted and added to \( \mathbf{D}_{t-1} \). We start from the initial depth map \( \mathbf{D}_0 \) at \( \frac{1}{8} \) resolution and obtain the depth map \( \mathbf{D}_N \) at the same resolution after \( N \) iterations.

2) Coarse-to-Fine Upsampling: The construction of correlation volumes and the iterative depth updating are performed at \( \frac{1}{8} \) resolution for efficiency. Then we design a depth upsampling module to recover object details in a coarse-to-fine manner by three \( 2 \times \) upsampling layers, as Fig. 4 shows.
### IV. Experiments

**A. Dataset and Metrics**

1) **Datasets:** Deep multi-view depth estimation methods choose different benchmarks for evaluation. DeepMVS [8], DPSNet [11], NAS [12] use DeMoN [19] and ETH3D [20] while DELTAS [42], CNM [57], EST [15] are evaluated on ScanNet [18], SUN3D [58], and 7Scenes [21]. In order to fully demonstrate the superiority of our method, we evaluate it on all datasets mentioned above. Since we focus on depth estimation rather than reconstruction, the MVS benchmarks DTU [59] and Tank & Templates [60] are not considered. ScanNet [18] contains more than 1600 indoor scenes with different environments, layouts and textures. Color images, ground-truth (GT) depth maps, camera intrinsics, and extrinsics are provided. We follow the ScanNet v2 official split to divide the training, validation, and test sets.

DeMoN is a mixed dataset introduced by Ummenhofer et al. [19] for multi-view depth estimation. Its data comes from MVS [46], SUN3D [58], RGBD [61], and Scenes11 [19]. In particular, MVS contains real-world outdoor environments, which is only used for evaluation. Scenes11 is a synthetic dataset generated by random shapes and motions whereas SUN3D and RGBD consist of real-world indoor environments. ETH3D [20] is a large-scale MVS dataset consisting of calibrated high-resolution images of indoor and outdoor challenge scenes with large viewpoint variations. Its test set is used in our paper for the evaluation of methods’ generalization ability. 7Scenes [21] is collected from 7 different indoor scenes. It consists of tracked RGB-D camera frames in which depth and RGB images are not aligned. Thus, we do not conduct a quantitative evaluation on 7Scenes. Instead, we reconstruct the scenes based on the predicted depth and the implementation of TSDF fusion [56].

2) **Evaluation Metrics:** We use five standard metrics for quantitative evaluation:

- **Absolute relative depth error (Abs Rel):** \( \frac{1}{N} \sum_{i=1}^{N} \frac{d_{gt} - d^i}{d_{gt}} \),
- **Absolute depth error (Abs):** \( \frac{1}{N} \sum_{i=1}^{N} \| d_{gt} - d^i \| \),
- **Absolute squared relative depth error (Abs Sq Rel):** \( \frac{1}{N} \sum_{i=1}^{N} \left( \frac{d_{gt} - d^i}{d_{gt}} \right)^2 \),
- **Root Mean Squared Error (RMSE):** \( \sqrt{\frac{1}{N} \sum_{i=1}^{N} (d_{gt} - d^i)^2} \),
- **Absolute depth error (Abs Sq Rel):** \( \frac{1}{N} \sum_{i=1}^{N} (d_{gt} - d^i)^2 \),
- **Absolute depth error (Abs Sq Rel):** \( \frac{1}{N} \sum_{i=1}^{N} \left( \frac{d_{gt} - d^i}{d_{gt}} \right)^2 \),
- **Absolute depth error (Abs):** \( \frac{1}{N} \sum_{i=1}^{N} \| d_{gt} - d^i \| \),
- **Absolute relative depth error (Abs Rel):** \( \frac{1}{N} \sum_{i=1}^{N} \frac{d_{gt} - d^i}{d_{gt}} \).

### Table I

| Method      | Abs Rel | Abs Sq Rel | RMSE | \( \delta < 1.25 \) | ScanNet | SUN3D* |
|-------------|---------|------------|------|----------------------|---------|--------|
| MVDepth [9] | 0.1167  | 0.2301     | 0.0596 | 0.3236 | 84.53   | 0.1377 | 0.3199 |
| MVDepth-FT  | 0.1116  | 0.2087     | 0.0763 | 0.3143 | 88.04   | 0.3092 | 0.7209 | 4.4899 | 1.7180 | 78.73 |
| DPS [11]    | 0.1200  | 0.2104     | 0.0688 | 0.3139 | 86.40   | 0.1469 | 0.3355 | 0.1165 | 0.4489 | 78.12 |
| DPS-FT      | 0.0986  | 0.1998     | 0.0459 | 0.2840 | 88.80   | 0.1274 | 0.2858 | 0.0855 | 0.3815 | 83.96 |
| NAS [12]    | 0.0941  | 0.1928     | 0.0417 | 0.2703 | 90.09   | 0.1271 | 0.2879 | 0.0852 | 0.3775 | 82.95 |
| DELTAS [42] | 0.0915  | 0.1710     | 0.0327 | 0.2390 | 91.47   | 0.1245 | 0.2662 | 0.0741 | 0.3602 | 85.51 |
| EST [15]    | 0.0812  | 0.1505     | 0.0298 | 0.2199 | 93.13   |        |        |        |        |        |
| Ours        | 0.0607  | 0.1162     | 0.0205 | 0.1915 | 95.99   | 0.1121 | 0.2552 | 0.0661 | 0.3369 | 87.15 |

*COMPARISON ON SCANNET AND SUN3D DATASET. OUR METHOD OUTPERFORMS OTHER METHODS BY A LARGE MARGIN. (* INDICATES THE DATASET IS NOT USED DURING TRAINING. EST REQUIRES MULTIPLE FRAMES TO EXTRACT TEMPORAL INFORMATION ON THE VIDEO, THUS CAN NOT BE DIRECTLY TESTED ON SUN3D WHICH ONLY PROVIDES TWO-VIEW IMAGE PAIRS.)*
### Table II

Evaluation on DeMoN and ETH3D. * indicates the datasets are not used during training. Hybrid means containing both indoor and outdoor scenes. Bold indicates the best and underline indicates the second.

| Method | SUN3D (Indoor) AbsRel $\delta < 1.25$ | RRGD (Indoor) AbsRel $\delta < 1.25$ | Scenes11 (Synthetic) AbsRel $\delta < 1.25$ | MVS* (Outdoor) AbsRel $\delta < 1.25$ | ETH3D* (Hybrid) AbsRel $\delta < 1.25$ |
|--------|----------------------------------------|----------------------------------------|----------------------------------------|----------------------------------------|----------------------------------------|
| COLMAP | 0.6232                                 | 0.5389                                 | 0.3841                                 | 0.324                                  | 86.5                                   |
| DeMoN  | 0.2137                                 | 0.1569                                 | 0.2305                                 | 0.178                                  | 85.8                                   |
| DeepMVS| 0.2816                                 | 0.2938                                 | 0.2305                                 | 0.178                                  | 85.8                                   |
| DPSNet | 0.1469                                 | 0.1508                                 | 0.0813                                 | 0.091                                  | 86.3                                   |
| NAS    | 0.1271                                 | 0.1314                                 | 0.0679                                 | 0.091                                  | 86.3                                   |
| Ours   | **0.1068**                             | **0.0943**                             | **0.0380**                             | **0.0388**                             | **88.6**                               |

Fig. 5. Qualitative comparison with other learning-based methods. Benefiting from our dense and accurate correspondences, our method recovers more object details with sharp boundaries (black box) and generates the most precise depth prediction, especially at textureless regions (red boxes).

- Square relative error (Sq Rel): $\frac{1}{M} \sum_{i=1}^{M} \frac{\|d^i_{gt} - d^i\|^2}{d^i_{gt}}$.
- Root mean square error (RMSE): $\sqrt{\frac{1}{M} \sum_{i=1}^{M} \|d^i_{gt} - d^i\|^2}$.
- The inlier ratio with threshold 1.25 ($\delta < 1.25$): $\delta = \max \left( \frac{d^i_{gt}}{d^i}, \frac{d^i}{d^i_{gt}} \right) < 1.25$.

where $M$ is the number of pixels that are valid in the depth ground truth while $d^i_{gt}$ and $d^i$ are the ground truth and predicted depth values for pixel $i$.

B. Implementation Details

We implement our model with PyTorch and use the AdamW optimizer [62]. The learning rate is 6e-4 and we clip gradients to the range $[-1, 1]$. Our correlation-guided depth refinement module is initialized from scratch with random weights while the feature encoder $\theta$ and optical flow estimation network is pretrained from RAFT. The threshold $r$ in the correlation fusion module is set to 3. The iteration times on the depth refinement stage is 12 for both training and testing. To evaluate our framework on the different benchmarks with other methods fairly, we train two independent models, one on ScanNet and the other on DeMoN. We train our model with the batch size of 48 on 8 NVIDIA V100 GPUs for 80k iterations. It takes about 36 hours for training. On both datasets, the input image resolution is 640 x 480. For ScanNet, we follow EST’s view selection strategy. For each reference image $I_k$, we choose its four neighboring frames $\{I_{k-2}, I_{k-1}, I_{k+1}, I_{k+2}\}$ with the interval of ten as the source images. For DeMoN, we follow its official train/test split as previous methods [11], [12].

C. Multi-View Depth Estimation Evaluation

Quantitative Evaluation. For ScanNet, we compare our method with five learning-based multi-view depth estima-
Fig. 6. TSDF reconstruction of scenes in the ScanNet (the first two rows) and 7Scenes (the last two rows) dataset. Our approach better recovers the shape details, such as the cabinet and door (red box). Our depth prediction leads to smoother and denser surfaces with the minimum distortion (green box) and fewer noises (blue box). Better viewed when zooming in.

Quantitative Evaluation. As shown in Table I, even our model is not trained on SUN3D, it still outperforms all the other methods, including those trained on SUN3D, i.e., MVDepth, DPS, and NAS. EST requires multiple frames as input to extract temporal relations, thus it cannot be directly tested on SUN3D which only provides two-view image pairs. As shown in Table II, our method also achieves the best performance on MVS and the comparable results on ETH3D with SOTA NAS. However, our method is more efficient (Fig. 7) and does not need additional normal map for supervision compared to NAS. The generalization ability of our framework comes from the general representation and interpretable usage of correspondence.

Qualitative Evaluation. As shown in the last two columns in Fig. 6, we reconstruct an office scene in unseen dataset 7Scenes by 50 images sampled every ten frames. Our approach better recovers the scene structure (top view) and object details.
(front view). Besides, the reconstructed surfaces are smoother and contain fewer noises compared with other methods. This further proves the generalization ability of our method and its superiority in reconstruction.

E. Efficiency Analysis

Our correlation-based depth estimation framework not only produces high-quality depth maps, but also costs lower computations and memory. In Fig. 7 we compare the number of network parameters and GPU memory usage of the model with the state-of-the-art learning-based multi-view depth estimation methods [11], [12], [15], [42]. These methods are tested with the same setting: one reference image with four source images at the resolution of 640 \times 480 on ScanNet. In comparison, our model is lightweight, which has the second-lowest number of parameters (11.5M) and GPU memory cost (2.778M) while ensuring the highest accuracy. Although DPS [11] has fewer parameters and memory costs than our model, our method outperforms it on accuracy by a large margin.

F. Ablation study

We conduct a set of experiments to demonstrate the impact of each component in our framework, including correlation fusion, iteration times, and context features. More ablation studies on the number of source views, depth initialization, and depth upsampling can be found in the supplementary material.

Correlation Fusion of Neighboring Points. As described in Sec. III-D1, we first construct a correlation map \( V_k \) between the reference image and a source image by reprojecting each pixel to the source view and concatenating the correlations of neighboring points. Another option is directly concatenating the correlations at a pixel in the correlation pyramid in the last two dimensions and applying 2D convolution, similar to the cost volume regularization in previous methods. As shown in Table III, this ‘Conv. Map’ does not perform well compared to our reprojectation strategy that establishes an explicit relationship between the depth prediction and correlation volumes.

Fusion Correlation of Multiple Views. When fusing the correlation maps \( \{ V_k \}_{k=1}^m \) with multiple source views, different fusion methods can be used, such as averaging, using their variance, max-pooling, or \( 1 \times 1 \) convolution. As Table III shows, ‘Convolution’ leads to the best results, showing its strong ability in learning effective fusion patterns. However, the convolution brings more parameters and requires a fixed number of input frames, which is not flexible enough. Similarly, the variance strategy is not suitable for two-view stereo.

Contrarily, the averaging and max-pooling operations work for arbitrary view numbers with slight accuracy loss. We finally select ‘Averaging’ because of its comparable results with max-pooling and better generalization ability on other datasets.

Iteration Times. For the iterative depth updating described in Sec. III-D1 we verify the convergence of the depth update value \( |\Delta d| \) and determine the optimal number of iterations considering the mean \( |\Delta d| \) of all training data and the absolute relative error of the test data. It shows that our iterative depth updating method progressively refines the initial depth map and converges in a few iterations. According to the trends of depth updating shown in Fig. 8 we select \( N = 12 \) iterations in our experiments. In addition, the time of depth refinement increases linearly with the iteration, which reaches about 115ms at the 12th iteration.

Context Features. While the correlation volumes mainly describe pixel-wise similarities between multiple views, the context feature extracted from the reference image provides more spatial context between neighboring points and serves as a regularization term during iterative updating. As shown in Fig. 9 it improves the depth estimation on low-texture regions and the non-overlapping regions when the views change greatly. We compare different implementations of the

---

**Fig. 7.** Efficiency comparison with other methods concerning depth prediction errors on ScanNet with model parameters (a) and memory cost (b).

**Fig. 8.** The iterative depth updates, the absolute relative error between the predicted depth and ground truth, and time cost at different iterations.

**Fig. 9.** Comparison of the model with or without context features. Context feature provide extra information for the textureless region or the region does not overlap between the reference image and target image.
context encoder $g$ in Table IV on ScanNet. We test a model (‘None’) without using context features for depth refinement. Since the optical flow estimation module in RAFT also employs the context feature, we test another variant of our model (‘Shared’) which directly uses the pretrained context encoder $g$ in RAFT for our depth refinement. In our final model, we train an independent context encoder $g$ for depth refinement. As shown in Table IV training an independent context encoder for depth refinement (‘Indep.’) leads to the best performance. However, even without using the context feature, our model ‘None’ still outperforms existing learning-based methods, demonstrating the effectiveness of correlation-based depth estimation.

V. Discussion and Limitation

The experiment results on different datasets demonstrate the superiority of our model by introducing a new way to represent and exploit pixel correspondences into multi-view depth estimation. Compared to the previous learning-based methods, we take advantage of deep learning and multi-view geometry to endow the model with stronger generalization capability. Furthermore, since pixel correspondence is a fundamental and generic medium representation, our framework can be extended to many other multi-view or video tasks, such as multi-view pose estimation, video tracking, etc. It is worth exploring the generality of the correlation volume among these vision tasks. Another direction is to explore whether joint learning can bring more robust pixel correspondence. However, our framework also has limitations. Currently, we have not made specific designs for dynamic objects that violate the depth-based reprojection principle. Adding semantic priors or estimating motion flows can be potential solutions for depth prediction in dynamic scenes.

VI. Conclusion

In this paper, we propose a novel multi-view depth estimation framework that fully exploits multi-view correlations to mimic traditional optimization process. Distinctive to cost-volume-based methods that require a set of predefined depth hypotheses, we directly infer dense point-wise correspondence for depth estimation from multi-view correlations. We design an correlation-guided depth refinement module that incrementally updates the depth prediction from image correlations. Our correlation fusion based on point reprojection and depth updating based on fused correlations effectively integrates multi-view geometry for structure recovery and DNNs for correlation fusion. Besides, the multi-scale correlations can alleviate mismatching on confusing areas by fusing sufficient neighboring information. The combination of DNNs and multi-view geometry endows our framework with stronger inferring ability and generalization capability. Experiments on ScanNet, DeMoN, ETH3D, and 7Scenes demonstrate that our model achieves state-of-the-art performance and is well generalized to various scenes.

References

[1] J. L. Schönberger, E. Zheng, J.-M. Frahm, and M. Pollefeys, “Pixel-wise view selection for unstructured multi-view stereo,” in European Conference on Computer Vision. Springer, 2016, pp. 501–518.
[2] S. Song and J. Xiao, “Sliding shapes for 3D object detection in depth images,” in European Conference on Computer Vision. Springer, 2014, pp. 634–651.
[3] C. Couprie, C. Farabet, L. Najman, and Y. LeCun, “Indoor semantic segmentation using depth information,” in First International Conference on Learning Representations (ICLR 2013), 2013, pp. 1–8.
[4] 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.
[5] R. T. Collins, “A space-sweep approach to true multi-image matching,” in Proceedings CVPR IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE, 1996, pp. 358–363.
[6] A. Hosni, C. Rhemann, M. Bleyer, C. Rother, and M. Gelautz, “Fast cost-volume filtering for visual correspondence and beyond,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 35, no. 2, pp. 504–511, 2012.
[7] H. Hirschmuller, “Stereo processing by semiglobal matching and mutual information,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 30, no. 2, pp. 328–341, 2007.
[8] P.-H. Huang, K. Matzen, J. Kopf, N. Ahuja, and J.-B. Huang, “DeepMVS: Learning multi-view stereopsis,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 2821–2830.
[9] K. Wang and S. Shen, “MVDiffNet: Real-time multi-view depth estimation neural network,” in 2018 International Conference on 3D Vision (3DV). IEEE, 2018, pp. 248–257.
[10] 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), 2018, pp. 767–783.
[11] S. Im, H.-G. Jeon, S. Lin, and I. S. Kweon, “DPSNet: End-to-end deep plane sweep stereo,” in International Conference on Learning Representations, 2018.
[12] U. Kusupati, S. Cheng, R. Chen, and H. Su, “Normal assisted stereo depth estimation,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 2189–2199.
[13] J. Yang, W. Mao, J. M. Alvarez, and M. Liu, “Cost volume pyramid based depth inference for multi-view stereo,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 4877–4886.
[14] A. Duzceker, S. Galliani, C. Vogel, P. Speciale, M. Dusmanu, and M. Pollefeys, “DeepVideoMVS: Multi-view stereo on video with recurrent spatio-temporal fusion,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 15 324–15 333.
[15] X. Long, L. Liu, W. Li, C. Theobalt, and W. Wang, “Multi-view depth estimation using epipolar spatio-temporal networks,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 8258–8267.
[16] Y. Furukawa and C. Hernández, “Multi-view stereo: A tutorial,” Foundations and Trends® in Computer Graphics and Vision, vol. 9, no. 1-2, pp. 1–148, 2015.
[17] Z. Teed and J. Deng, “RAFT: Recurrent all-pairs field transforms for optical flow,” in European Conference on Computer Vision. Springer, 2020, pp. 402–419.
[18] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nielßen, “ScanNet: Richly-annotated 3D reconstructions of indoor scenes,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 5828–5837.
[19] B. Ummenhofer, H. Zhou, J. Uhrig, N. Mayer, E. Bg, A. Dosovitskiy, and T. Brox, “DeMoN: Depth and motion network for learning monocular stereo,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 5038–5047.
[20] 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.

[21] J. Shotton, B. Glocker, C. Zach, S. Izadi, A. Criminisi, and A. Fitzgibbon, “Scene coordinate regression forests for camera relocalization in RGB-D images,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 2930–2937.

[22] Y. Furukawa and J. Ponce, “Accurate, dense, and robust multi-view stereopsis,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 8, pp. 1362–1376, 2009.

[23] M. Lhuillery 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.

[24] 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.

[25] 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.

[26] J. L. Schonberger and J.-M. Frahm, “Structure-from-motion revisited,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 4104–4113.

[27] P. Lindenberger, P.-E. Sarlin, V. Larsson, and M. Pollefeys, “Pixel-perfect structure-from-motion with featuremetric refinement,” in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 5987–5997.

[28] R. Mur-Artal and J. D. Tardós, “ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-D cameras,” IEEE Transactions on Robotics, vol. 33, no. 5, pp. 1255–1262, 2017.

[29] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardós, “ORB-SLAM: A versatile and accurate monocular SLAM system,” IEEE Transactions on Robotics, vol. 31, no. 5, pp. 1147–1163, 2015.

[30] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” International Journal of Computer Vision, vol. 60, no. 2, pp. 91–110, 2004.

[31] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, “ORB: An efficient alternative to SIFT or SURF,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2011, pp. 224–231.

[32] D. DeTone, T. Malisiewicz, and A. Rabinovich, “Superpoint: Self-supervised interest point detection and description,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 224–231.

[33] X. Liu, Y. Zheng, B. Killeen, M. Ishii, G. D. Hager, R. H. Taylor, and M. Umberath, “Extremely dense point correspondences using a learned feature descriptor,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 4847–4856.

[34] Q. Yang, L. Wang, R. Yang, H. Stewénius, and D. Nistér, “Stereo matching with color-weighted correlation, hierarchical belief propagation, and occlusion handling,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 3, pp. 492–504, 2017.

[35] T. Kanade and M. Okutomi, “A stereo matching algorithm with an adaptive window: Theory and experiment,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 16, no. 9, pp. 920–932, 1994.

[36] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. Van Der Smagt, D. Cremers, and T. Brox, “FlowNet: Optical flow with convolutional networks,” in Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 2758–2766.

[37] X. Long, L. Liu, C. Theobalt, and W. Wang, “Occlusion-aware depth estimation with adaptive normal constraints,” in European Conference on Computer Vision. Springer, 2020, pp. 640–657.

[38] J. Xiao, A. Owens, and A. Torralba, “SUN3D: A database of big spaces reconstructed using 2D and object labels,” in Proceedings of the IEEE International Conference on Computer Vision, 2013, pp. 1625–1632.

[39] 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.

[40] A. Knipfisch, J. Park, Q.-Y. Zhou, and V. Koltun, “Tanks and Temples: Benchmarking large-scale multi-view reconstruction,” ACM Transactions on Graphics, vol. 36, no. 4, 2017.

[41] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, “A benchmark for the evaluation of RGB-D SLAM systems,” in 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, 2012, pp. 573–580.

[42] I. Loshchilov and F. Hutter, “Decoupled weight decay regularization,” in International Conference on Learning Representations, 2018.

[43] Procedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 5525–5534.

[44] A. Sinha, Z. Murez, J. Bartolozzi, V. Badrinarayanan, and A. Roninovich, “DELTAS: Depth estimation by learning triangulation and densification of sparse points,” in Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, vol. 2, Part XXI, pp. 104–121.

[45] K. Luo, Y. Zhang, D. Yu, L. Sun, and D. Nistér, “DELTAS: Depth estimation by learning triangulation and object labels,” in Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, 2020, pp. 4312–4321.

[46] 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, 2019, pp. 5525–5534.