DeepSFM: Structure From Motion Via Deep Bundle Adjustment

Xingkui Wei[1*], Yinda Zhang[2*], Zhuwen Li[3*], Yanwei Fu[1 †], and Xiangyang Xue[1]

[1]Fudan University [2]Google Research [3]Nuro, Inc

Abstract. Structure from motion (SfM) is an essential computer vision problem which has not been well handled by deep learning. One of the promising trends is to apply explicit structural constraint, e.g. 3D cost volume, into the network. However, existing methods usually assume accurate camera poses either from GT or other methods, which is unrealistic in practice. In this work, we design a physical driven architecture, namely DeepSFM, inspired by traditional Bundle Adjustment (BA), which consists of two cost volume based architectures for depth and pose estimation respectively, iteratively running to improve both. The explicit constraints on both depth (structure) and pose (motion), when combined with the learning components, bring the merit from both traditional BA and emerging deep learning technology. Extensive experiments on various datasets show that our model achieves the state-of-the-art performance on both depth and pose estimation with superior robustness against less number of inputs and the noise in initialization.

1 Introduction

SfM is a fundamental human vision functionality which recovers 3D structures from the projected retinal images of moving objects or scenes. It enables machines to sense and understand the 3D world and is critical in achieving real-world artificial intelligence. Over decades of researches, there has been a lot of great success on SfM; however, the performance is far from perfect.

Conventional SfM approaches [1,46,8,5] heavily rely on Bundle-Adjustment (BA) [40,2], in which 3D structures and camera motions of each view are jointly optimized via Levenberg-Marquardt (LM) algorithm [32] according to the cross-view correspondence. Though successful in certain scenarios, conventional SfM based approaches are fundamentally restricted by the coverage of the provided multiple views and the overlaps among them. They also typically fail to reconstruct textureless or non-lambertian (e.g. reflective or transparent) surfaces due to the missing of correspondence across views. As a result, selecting sufficiently good input views and the right scene requires excessive caution and is usually non-trivial to even experienced user.

*indicates equal contributions.

†indicates corresponding author.

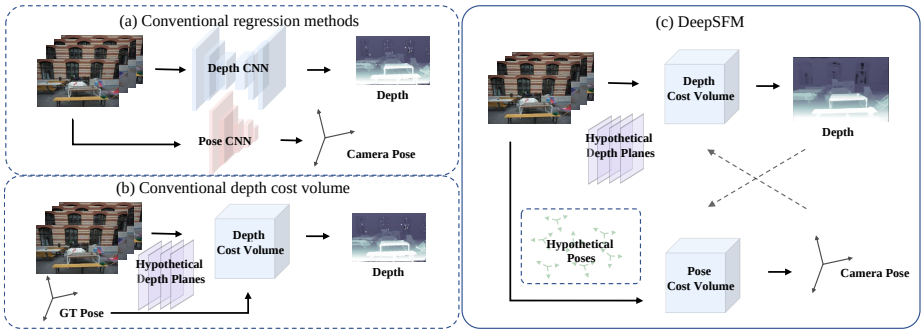


Fig. 1. DeepSFM refines the depth and camera pose of a target image given a few nearby source images. The network includes a depth based cost volume (D-CV) and a pose based cost volume (P-CV) which enforce photo-consistency and geometric-consistency into 3D cost volumes. The whole procedure is performed as iterations.

Recent researches resort to deep learning to deal with the typical weakness of conventional SfM. Early effort utilizes deep neural network as a powerful mapping function that directly regresses the structures and motions [41,42,52,45]. Since the geometric constraints of structures and motions are not explicitly enforced, the network does not learn the underlying physics and prone to overfitting. Consequently, they do not perform as accurate as conventional SfM approaches and suffer from extremely poor generalization capability. Most recently, the 3D cost volume [22] has been introduced to explicit leveraging photo-consistency in a differentiable way, which significantly boosts the performance of deep learning based 3D reconstruction. However, the camera motion usually has to be known [50,20], which requires to run traditional methods on densely captured high resolution images or relies on extra calibration devices (Fig. 1 (b)). Some methods direct regress the motion [41,52], which still suffer from generalization issue (Fig. 1 (a)). Very rare deep learning approaches [38,39] can work well under noisy camera motion and improve both structure and motion simultaneously.

Inspired by BA and the success of cost volume for depth estimation, we propose a deep learning framework for SfM that iteratively improves both depth and camera pose according to cost volume explicitly built to measure photo-consistency and geometric-consistency. Our method does not require accurate pose, and a rough estimation is enough. In particular, our network includes a depth based cost volume (D-CV) and a pose based cost volume (P-CV). D-CV optimizes per-pixel depth values with the current camera poses, while P-CV optimizes camera poses with the current depth estimations (see Fig.1 (c)). Conventional 3D cost volume enforces photo-consistency by unprojecting pixels into the discrete camera fronto-parallel planes and computing the photometric (i.e. image feature) difference as the cost. In addition to that, our D-CV further enforces geometric-consistency among cameras with their current depth estimations by adding the geometric (i.e. depth) difference to the cost. Note that the

initial depth estimation can be obtained using the conventional 3D cost volume. When preparing this work, we notice that a concurrent work [49] which also utilizes this trick to build a better cost volume in their system. For pose estimation, rather than direct regression, our P-CV discretizes around the current camera positions, and also computes the photometric and geometric differences by hypothetically moving the camera into the discretized position. Note that the initial camera pose can be obtained by a rough estimation from the direct regression methods such as [41]. Our framework bridges the gap between the conventional and deep learning based SfM by incorporating explicit constraints of photo-consistency, geometric-consistency and camera motions all in the deep network.

The closest work in the literature is the recently proposed BA-Net [38], which also aims to explicitly incorporate multi-view geometric constraints in a deep learning framework. They achieve this goal by integrating the LM optimization into the network. However, the LM iterations are unrolled with few iterations due to the memory and computational inefficiency, and thus it can potentially lead to non-optimal solutions due to lack of enough iterations. In contrast, our method does not have a restriction on the number of iterations and achieves empirically better performance. Furthermore, LM in SfM originally optimizes point and camera positions, and thus direct integration of LM still requires good correspondences. To evade the correspondence issue in typical SfM, their models employ a direct regressor to predict depth at the front end, which heavily relies on prior in the training data. In contrast, our model is a fully physical-driven architecture that less suffers from over-fitting issue for both depth and pose estimation.

To demonstrate the superiority of our method, we conduct extensive experiments on *DeMoN datasets*, *ScanNet*, *ETH3D* and *Tanks and Temples*. The experiments show that our approach outperforms the state-of-the-art [33,41,38].

2 Related work

There is a large body of work that focuses on inferring depth or motion from color images, ranging from single view, multiple views and monocular video. We discuss them in the context of our work.

Single-view Depth Estimation. While ill-posed, the emerging of deep learning technology enables the estimation of depth from a single color image. The early work directly formulates this into a per-pixel regression problem [7], and follow-up works improve the performance by introducing multi-scale network architectures [7,6], skip-connections [44,26], powerful decoder and post process [13,25,24,44,26], and new loss functions [10]. Even though single view based methods generate plausible results, the models usually resort heavily to the prior in the training data and suffer from generalization capability. Nevertheless, these methods still act as an important component in some multi-view systems [38].

Traditional Structure-from-Motion Simultaneously estimating 3d structure and camera motion is a well studied problem which has a traditional

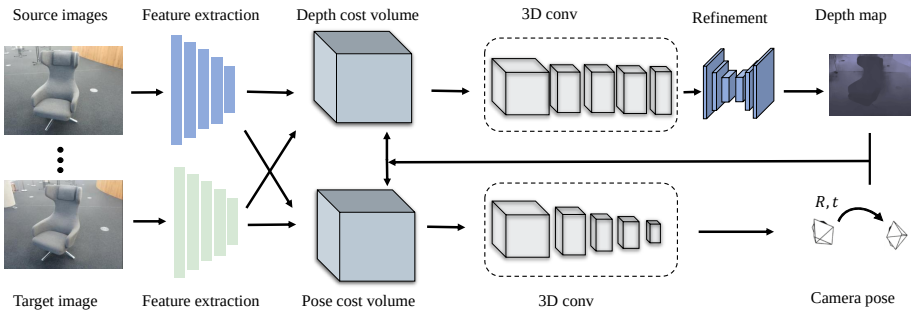


Fig. 2. Overview of DeepSFM. 2D CNN is used to extract photometric feature to construct cost volumes. Initial source depth maps and camera poses are used to introduce both photometric and geometric consistency. A series of 3D CNN layers are applied for D-CV and P-CV. Then a context network and depth regression operation are applied to produce predicted depth map of target image.

tool-chain of techniques [12,31,47]. Structure from Motion(SfM) has made great progress in many aspects. [28,15] aim at improving features and [35] introduce new optimization techniques. More robust structures and data representations are introduced by [14,33]. Simultaneous Localization and Sapping(SLAM) systems track the motion of the camera and build 3D structure from video sequence [31,9,29,30]. [9] propose the photometric bundle adjustment algorithm to directly minimize the photometric error of aligned pixels. However, traditional SfM and SLAM methods are sensitive to low texture region, occlusions, moving objects and lighting changes, which limit the performance and stability.

Deep Learning for Structure-from-Motion Deep neural networks have shown great success in stereo matching and Structure-from-Motion problems. [41,45,42,52] regress depth and camera pose directly in a supervised manner or by introducing photometric constraints between depth and motion as a self-supervision signal. Such methods solve the camera motion as a regression problem, and the relation between camera motion and depth prediction is neglected.

Recently, some methods exploit multi-view photometric or feature-metric constraints to enforce the relationship between dense depth and the camera pose in network. The SE3 transformer layer is introduced by [39], which uses geometry to map flow and depth into a camera pose update. [43] propose the differentiable camera motion estimator based on the Direct Visual Odometry [36]. [4] using a LSTM-RNN [18] as the optimizer to solve nonlinear least squares in two-view SfM. [38] train a network to generate a set of basis depth maps and optimize depth and camera poses in a BA-layer by minimizing a feature-metric error.

3 Architecture

Our framework receives frames of a scene from different viewpoints, and produces accurate depth maps and camera poses for all frames. Similar to Bundle

Adjustment (BA), we also assume initial structures (i.e depth maps) and motions (i.e. camera poses) are given. The initialization is not necessary to be accurate for the good performance using our framework and thus can be easily obtained from some direct regression based methods [41].

Now we introduce the overview of our model – DeepSFM. Without loss of generality, we describe our model taking two images as inputs, namely the target image and the source image, and all the technical components can be extended for multiple images straightforwardly. As shown in Fig.2, we first extract feature maps from inputs through a shared encoder. We then sample the solution space for depth uniformly in the inverse-depth space between a predefined range, and camera pose around the initialization respectively. After that, we build cost volumes accordingly to reason the confidence of each depth and pose hypothesis. This is achieved by validating the consistency between the feature of the target view and the ones warped from the source image. Besides photo-metric consistency that measures the color image similarity, we also take into account the geometric consistency across warped depth maps. Note that depth and pose require different designs of cost volume to efficiently sample the hypothesis space. Gradients can back-propagate through cost volumes, and cost-volume construction does not affect any trainable parameters. The cost volumes are then fed into 3D CNN to regress new depth and pose. These updated values can be used to create new cost volumes, and the model improves the prediction iteratively.

For notations, we denote $\{\mathbf{I}_i\}_{i=1}^n$ as all the images in one scene, $\{\mathbf{D}_i\}_{i=1}^n$ as the corresponding ground truth depth maps, $\{\mathbf{K}_i\}_{i=1}^n$ as the camera intrinsics, $\{\mathbf{R}_i, \mathbf{t}_i\}_{i=1}^n$ as the ground truth rotations and translations of camera, $\{\mathbf{D}_i^*\}_{i=1}^n$ and $\{\mathbf{R}_i^*, \mathbf{t}_i^*\}_{i=1}^n$ as initial depth maps and camera pose parameters for constructing cost volumes, where n is the number of image samples.

3.1 2D Feature Extraction

Given the input sequences $\{\mathbf{I}_i\}_{i=1}^n$, we extract the 2D CNN feature $\{\mathbf{F}_i\}_{i=1}^n$ for each frame. Firstly, a 7 layers' CNN with kernel size 3×3 is applied to extract low contextual information. Then we adopt a spatial pyramid pooling (SPP) [21] module, which can extract hierarchical multi-scale features through 4 average pooling blocks with different pooling kernel size $(4 \times 4, 8 \times 8, 16 \times 16, 32 \times 32)$. Finally, we pass the concatenated features through 2D CNNs to get the 32-channel image features after upsampling these multi-scale features into the same resolution. These image sequence features are used by the building of both our depth based and pose based cost volumes.

3.2 Depth based Cost Volume (D-CV)

Traditional plane sweep cost volume aims to back-project the source images onto successive virtual planes in the 3D space and measure photo-consistency error among the warped image features and target image features for each pixel. Different from the cost volume used in mainstream multi-view and structure-from-motion methods, we construct a D-CV to further utilize the local geometric

consistency constraints introduced by depth maps. Inspired by the traditional plane sweep cost volumes, our D-CV is a concatenation of three components: the target image features, the warped source image features and the homogeneous depth consistency maps.

Hypothesis Sampling To back-project the features and depth maps from source viewpoint to the 3D space in target viewpoint, we uniformly sample a set of L virtual planes $\{d_l\}_{l=1}^{L}$ in the inverse-depth space which are perpendicular to the forward direction (z-axis) of the target viewpoint. These planes serve as the hypothesis of the output depth map, and the cost volume can be built upon them.

Feature warping To construct our D-CV, we first warp source image features $\mathbf{F}_i$ (of size $CHannel \times Width \times Height$) to each of the hypothetical depth map planes d_l using camera intrinsic matrix $\mathbf{K}$ and initial camera poses $\{\mathbf{R}_i^*, \mathbf{t}_i^*\}$, according to:

$$\tilde{\mathbf{F}}_{il}(u) = \mathbf{F}_i\left(\tilde{u}_l\right), \tilde{u}_l \sim \mathbf{K}\left[\mathbf{R}_i^* | \mathbf{t}_i^*\right] \begin{bmatrix} \left(\mathbf{K}^{-1} u\right) d_l \\ 1 \end{bmatrix} \tag{1}$$

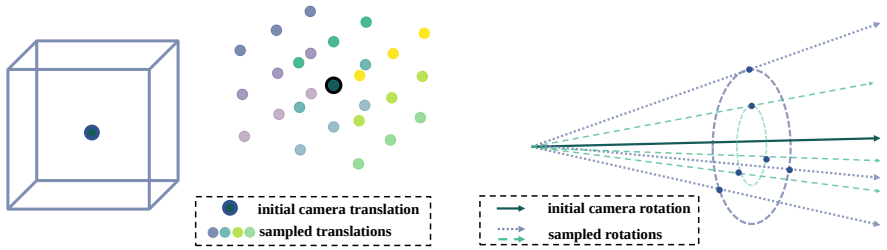
where u and $\tilde{u}_l$ are the homogeneous coordinates of each pixel in the target view and the projected coordinates onto the corresponding source view. $\tilde{\mathbf{F}}_{il}(u)$ denotes the warped feature of the source image through the l-th virtual depth plane. Note that the projected homogeneous coordinates $\tilde{u}_l$ are floating numbers, and we adopt a differentiable bilinear interpolation to generate the warped feature map $\tilde{\mathbf{F}}_{il}$. The pixels with no source view coverage are assigned with zeros. Following [20], we concatenate the target feature and the warped target feature together and obtain a $2CH \times L \times W \times H$ 4D feature volume.

Depth consistency In addition to photometric consistency, to exploit geometric consistency and promote the quality of depth prediction, we add two more channels on each virtual plane: the warped initial depth maps from the source views and the projected virtual depth plane from the perspective of the source view. Note that the former is the same as image feature warping, while the latter requires a coordinate transformation from the target to the source camera.

In particular, the first channel is computed as follows. The initial depth map of source image is first down-sampled and then warped to hypothetical depth planes similarly to the image feature warping as $\tilde{\mathbf{D}}_{il}^*(u) = \mathbf{D}_i^*\left(\tilde{u}_l\right)$, where the coordinates u and $\tilde{u}_l$ are defined in Eq. 1 and $\tilde{\mathbf{D}}_{il}^*(u)$ represents the warped one-channel depth map on the l-th depth plane. One distinction between depth warping and feature warping is that we adopt nearest neighbor sampling for depth warping, instead of bilinear interpolation. A comparison between the two methods is provided in the supplementary material.

The second channel contains the depth values of the virtual planes in the target view by seeing them from the source view. To transform the virtual planes to the source view coordinate system, we apply a T function on each virtual plane d_l in the following:

$$T(d_l) \sim \left[\mathbf{R}_i^* | \mathbf{t}_i^*\right] \begin{bmatrix} \left(\mathbf{K}^{-1} u\right) d_l \\ 1 \end{bmatrix} \tag{2}$$



(a) Camera translation sampling (b) Camera rotation sampling

Fig. 3. Hypothetical camera pose sampling. (a) Camera translation sampling. We sample uniformly in the cubic space. (b) Camera rotation sampling. We sample around initial orientation vector in conical space.

We stack the warped initial depth maps and the transformed depth planes together, and get a depth volume of size $2 \times L \times W \times H$.

By concatenating the feature volume and depth volume together, we obtain a 4D cost tensor of size $(2CH + 2) \times L \times W \times H$. Given the 4D cost volume, our network learns a cost volume of size $L \times W \times H$ using several 3D convolutional layers with kernel size $3 \times 3 \times 3$. When there is more than one source image, we get the final cost volume by averaging over multiple input source views.

3.3 Pose based Cost Volume (P-CV)

In addition to the construction of D-CV, we also propose a P-CV, aiming at optimizing initial camera poses through both photometric and geometric consistency (see Fig.3). Instead of building a cost volume based on hypothetical depth map planes, our novel P-CV is constructed based on a set of assumptive camera poses. Similar to D-CV, P-CV is also concatenated by three components: the target image features, the warped source image features and the homogeneous depth consistency maps. Given initial camera pose parameters $\{\mathbf{R}_i^*, \mathbf{t}_i^*\}$, we uniformly sample a batch of discrete candidate camera poses around. As shown in Fig.3, we shift rotation and translation separately while keeping the other one unchanged. For rotation, we sample δR uniformly in the Euler angle space in a predefined range and multiply δR by the initial R. For translation, we sample δt uniformly and add δt to the initial t. In the end, a group of P virtual camera poses noted as $\{\mathbf{R}_{ip}^*|\mathbf{t}_{ip}^*\}_{p=1}^P$ around input pose are obtained for cost volume construction.

The posed-based cost volume is also constructed by concatenating image features and homogeneous depth maps. However, source view features and depth maps are warped based on sampled camera poses. For feature warping, we compute $\tilde{u}_p$ as following equations:

$$\tilde{u}_p \sim \mathbf{K} \left[\mathbf{R}_{ip}^*|\mathbf{t}_{ip}^*\right] \begin{bmatrix} \left(\mathbf{K}^{-1}u\right) \mathbf{D}_i^* \\ 1 \end{bmatrix} \tag{3}$$

where $\mathbf{D}_i^*$ is the initial target view depth. Similar to D-CV, we get warped source feature map $\tilde{\mathbf{F}}_{ip}$ after bilinear sampling and concatenate it with target view feature map. We also transform the initial target view depth and source view depth into one homogeneous coordinate system, which enhances the geometric consistency between camera pose and multi view depth maps.

After concatenating the above feature maps and depth maps together, we again build a 4D cost volume of size $(2CH + 2) \times P \times W \times H$, where W and H are the width and height of feature map, CH is the number of channels. We get output of size $1 \times P \times 1 \times 1$ from the above 4-D tensor after eight 3D convolutional layers with kernel size $3 \times 3 \times 3$, three 3D average pooling layers with stride size $2 \times 2 \times 1$ and one global average pooling at the end.

3.4 Cost Aggregation and Regression

For depth prediction, we follow the cost aggregation technique introduced by [20]. We adopt a context network, which takes target image features and each slice of the coarse cost volume after 3D convolution as input and produce the refined cost slice. The final aggregated depth based volume is obtained by adding coarse and refined cost slices together. The last step to get depth prediction of target image is depth regression by soft-argmax as proposed in [20]. For camera poses prediction, we also apply a soft-argmax function on pose cost volume and get the estimated output rotation and translation vectors.

3.5 Training

The DeepSFM learns the feature extractor, 3D convolution, and the regression layers in a supervised way. We denote $\hat{\mathbf{R}}_i$ and $\hat{\mathbf{t}}_i$ as predicted rotation angles and translation vectors of camera pose. Then the pose loss $\mathcal{L}_{rotation}$ is defined as the $L1$ distance between prediction and groundtruth. We denote $\hat{D}_i^0$ and $\hat{D}_i$ as predicted coarse depth map and refined depth map, then the depth loss function is defined as $\mathcal{L}_{depth} = \sum_i \lambda H(\hat{D}_i^0, \mathbf{D}_i) + H(\hat{D}_i, \mathbf{D}_i)$, where λ is weight parameter and function H is Huber loss. Our final objective $\mathcal{L}_{final} = \lambda_r \mathcal{L}_{rotation} + \lambda_t \mathcal{L}_{translation} + \lambda_d \mathcal{L}_{depth}$. The λs are determined empirically, and are listed in the supplementary material.

The initial depth maps and camera poses are obtained from DeMoN. To keep correct scale, we multiply translation vectors and depth maps by the norm of the ground truth camera translation. The whole training and testing procedure are performed as four iterations. During each iteration, we take the predicted depth maps and camera poses of previous iteration as new initialization. More details are provided in the supplementary material.

4 Experiments

4.1 Datasets

We evaluate DeepSFM on widely used datasets and compare with state-of-the-art methods on accuracy, generalization capability and robustness to initialization.

DeMoN Datasets [41] This dataset contains data from various sources, including SUN3D [48], RGB-D SLAM [37], and Scenes11 [3]. To test the generalization capability, we also evaluate on MVS [11] dataset but not use it for the training. In all four datasets, RGB image sequences and the ground truth depth maps are provided with the camera intrinsics and camera poses. Note that those datasets together provide a diverse set of both indoor and outdoor, synthetic and real-world scenes. For all the experiments, we adopt the same training and testing data split from DeMoN.

ETH3D Dataset [34] It provides a variety of indoor and outdoor scenes with high-precision ground truth 3D points captured by laser scanners, which is a more solid benchmark dataset. Ground truth depth maps are obtained by projecting the point clouds to each camera view. Raw images are in high resolution but resized to 810×540 pixels for evaluation [20].

Tanks and Temples [23] It is a benchmark for image-based large scale 3D reconstruction. The benchmark sequences are acquired in realistic conditions and of high quality. Point clouds captured using an industrial laser scanner are provided as ground truth. Again, our method are trained on DeMoN and tested on the dataset to show the robustness to noisy initialization.

4.2 Evaluation

DeMoN Datasets Our results on DeMoN datasets and the comparison to other methods are shown in Table 1. We cite results of some strong baseline methods from DeMoN paper, named as Base-Oracle, Base-SIFT, Base-FF and Base-Matlab respectively [41]. Base-Oracle estimate depth with the ground truth camera motion using SGM [17]. Base-SIFT, Base-FF and Base-Matlab solve camera motion and depth using feature, optical flow, and KLT tracking correspondence from 8-pt algorithm [16]. We also compare to some most recent state-of-the-art methods LS-Net [4] and BA-Net [38]. LS-Net introduces the learned LSTM-RNN optimizer to minimizing photometric error for stereo reconstruction. BA-Net is the most recent work that minimizes the feature-metric error between multi-view via the differentiable Levenberg-Marquardt [27] algorithm. To make a fair comparison, we adopt the same metrics as DeMoN[41] for evaluation.

Our method outperforms all traditional baseline methods and DeMoN on both depth and camera poses. When compared with more recent LS-Net and BA-Net, our method produces better results in most metrics on four datasets. On RGB-D dataset, our performance is comparable to the state-of-the-art due to relatively higher noise in the RGB-D ground truth. LS-Net trains an initialization network which regresses depth and motion directly before adding the

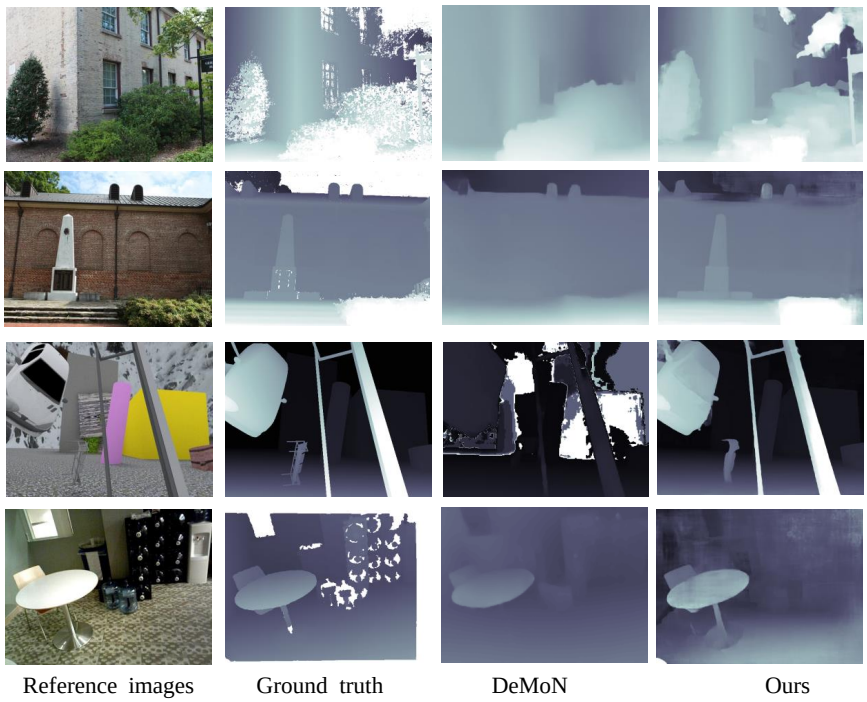


Fig. 4. Qualitative Comparisons with DeMoN [41] on DeMoN datasets. Results on more methods and examples are shown in the supplementary material.

LSTM-RNN optimizer. The performance of the RNN optimizer is highly affected by the accuracy of the regressed initialization. The depth results of LS-Net are consistently poorer than BA-Net and our method, despite better rotation parameters are estimated by LS-Net on RGB-D and Sun3D datasets with very good initialization. Our method is slightly inferior to BA-Net on the L1-rel metric, which is probably due to that we sample 64 virtual planes uniformly as the hypothetical depth set, while BA-Net optimizes depth prediction based on a set of 128-channel estimated basis depth maps that are more memory consuming but have more fine-grained results empirically. Despite all that, it is shown that our learned cost volumes with geometric consistency work better than the photometric bundle adjustment (e.g. used in BA-Net) in most scenes. In particular, we improve mostly on the Scenes11 dataset, where the ground truth is perfect but the input images contain a lot of texture-less regions, which are challenging to photo-consistency based methods. The Qualitative Comparisons between our method and DeMoN are shown in Fig.4.

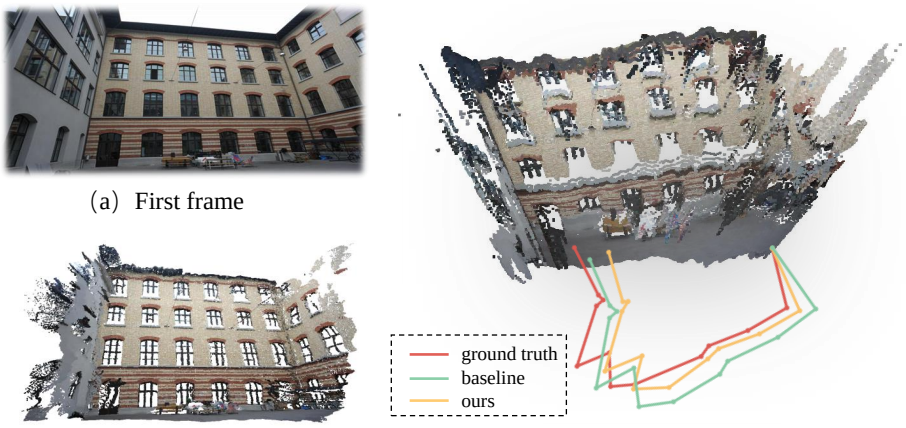
ETH3D We further test the generalization capability on ETH3D. We provide comparisons to COLMAP [33] and DeMoN on ETH3D. COLMAP is a state-of-the-art Structure-from-Motion method, while DeMoN introduces a classical deep network architecture that directly regress depth and motion in a super-

Table 1. Results on DeMoN datasets, the best results are noted by **Bold**.

MVS	Depth			Motion		**Scenes11**	Depth			Motion	
Method	L1-inv	sc-inv	L1-rel	Rot	Trans	Method	L1-inv	sc-inv	L1-rel	Rot	Trans
Base-Oracle	0.019	0.197	0.105	0	0	Base-Oracle	0.023	0.618	0.349	0	0
Base-SIFT	0.056	0.309	0.361	21.180	60.516	Base-SIFT	0.051	0.900	1.027	6.179	56.650
Base-FF	0.055	0.308	0.322	4.834	17.252	Base-FF	0.038	0.793	0.776	1.309	19.426
Base-Matlab	-	-	-	10.843	32.736	Base-Matlab	-	-	-	0.917	14.639
DeMoN	0.047	0.202	0.305	5.156	14.447	DeMoN	0.019	0.315	0.248	0.809	8.918
LS-Net	0.051	0.221	0.311	4.653	11.221	LS-Net	0.010	0.410	0.210	4.653	8.210
BANet	0.030	0.150	0.080	3.499	11.238	BANet	0.080	0.210	0.130	3.499	10.370
Ours	**0.021**	**0.129**	**0.079**	**2.824**	**9.881**	Ours	**0.007**	**0.112**	**0.064**	**0.403**	**5.828**
RGB-D	Depth			Motion		**Sun3D**	Depth			Motion	
Method	L1-inv	sc-inv	L1-rel	Rot	Trans	Method	L1-inv	sc-inv	L1-rel	Rot	Trans
Base-Oracle	0.026	0.398	0.36	0	0	Base-Oracle	0.020	0.241	0.220	0	0
Base-SIFT	0.050	0.577	0.703	12.010	56.021	Base-SIFT	0.029	0.290	0.286	7.702	41.825
Base-FF	0.045	0.548	0.613	4.709	46.058	Base-FF	0.029	0.284	0.297	3.681	33.301
Base-Matlab	-	-	-	12.813	49.612	Base-Matlab	-	-	-	5.920	32.298
DeMoN	0.028	0.130	0.212	2.641	20.585	DeMoN	0.019	0.114	0.172	1.801	18.811
LS-Net	0.019	0.090	0.301	**1.010**	22.100	LS-Net	0.015	0.189	0.650	**1.521**	14.347
BANet	**0.008**	0.087	**0.050**	2.459	14.900	BANet	0.015	0.110	**0.060**	1.729	13.260
Ours	0.011	**0.071**	0.126	1.862	**14.570**	Ours	**0.013**	**0.093**	0.072	1.704	**13.107**

vised manner. Note that all the models are trained on DeMoN and then tested on the data provided by [19]. In the accuracy metric, the error δ s defined as $\max(\frac{y_i^*}{y_i}, \frac{y_i}{y_i^*})$, and the thresholds are typically set as $[1.25, 1.25^2, 1.25^3]$. In Table 2, our method shows the best performance overall among all the comparison methods. Our method produces better results than DeMoN consistently, since we impose geometric and physical constraints onto network rather than learning to regress directly. When compared with COLMAP, our method performs better on most metrics. COLMAP behaves well in the accuracy metric (i.e. abs_diff). However, the presence of outliers is often observed in the predictions of COLMAP, which leads to poor performance in other metrics such as abs_rel and sq_rel, since those metrics are sensitive to outliers. As an intuitive display, we compute the motion of camera in a selected image sequence of ETH3D, as shown in Fig. 5c. The point cloud computed from the estimated depth map is showed in Fig.5b, which is of good quality.

Tanks and Temples To evaluate the robustness to initialization quality, we compare DeepSFM with COLMAP and the SOTA – R-MVSNet[51] on the Tanks and Temples[23] dataset as it contains densely captured high resolution images from which pose can be precisely estimated. To add noise on pose, we downscale the images and sub-sample temporal frames. For evaluation metrics, we adopt the F-score (higher is better) used in this dataset. The reconstruction



(a) First frame (b) Point cloud of first frame (c) Pose trajectory

Fig. 5. Result on a sequence of the ETH3D dataset. (a) First frame of sequence. (b) The point cloud from estimated depth of the first frame. (c) Pose trajectories. Compared with the baseline method in sec.4.3, the accumulated pairwise pose trajectory predicted by our network (yellow) are more closely consistent with the ground truth (red).

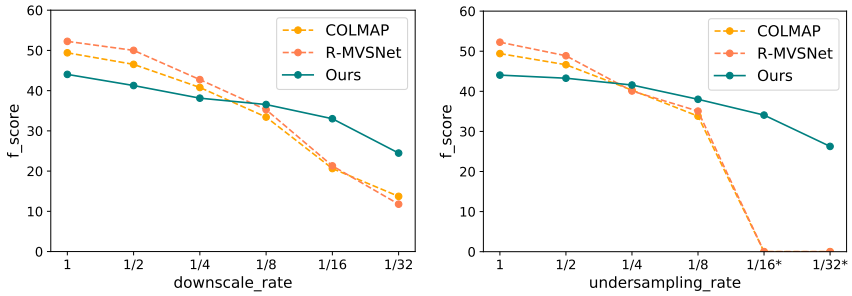
qualities of Barn sequence are shown in fig.6. It is observed that the performance of R-MVSNet and COLMAP drops significantly as the input quality becomes lower, while our method maintains the performance in a certain range. It is worth noting that COLMAP completely fails when the number of images are sub-sampled to 1/16.

4.3 Model Analysis

In this section, we analyze our model on several aspects to verify the optimality and show advantages over previous methods. More ablation studies are provided in the supplementary material.

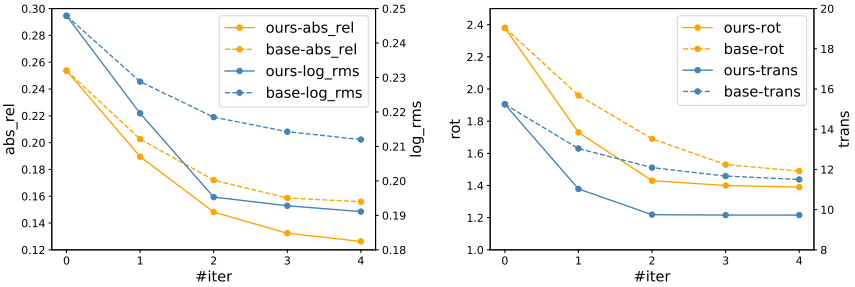
Table 2. Results on ETH3D (**Bold**: best; $\alpha = 1.25$). abs_rel, abs_diff, sq_rel, rms, and log_rms, are absolute relative error, absolute difference, square relative difference, root mean square and log root mean square, respectively.

Method	Error metric					Accuracy metric($\delta < \alpha^t$)		
	abs_rel	abs_diff	sq_rel	rms	log_rms	α	α^2	α^3
COLMAP	0.324	**0.615**	36.71	2.370	0.349	**86.5**	90.3	92.7
DeMoN	0.191	0.726	0.365	1.059	0.240	73.3	89.8	95.1
Ours	**0.127**	0.661	**0.278**	**1.003**	**0.195**	84.1	**93.8**	**96.9**



(a) resolution downscale (b) temporary sub-sample

Fig. 6. Comparison with COLMAP[33] and R-MVSNet[51] with noisy input. Our work is less sensitive to initialization.

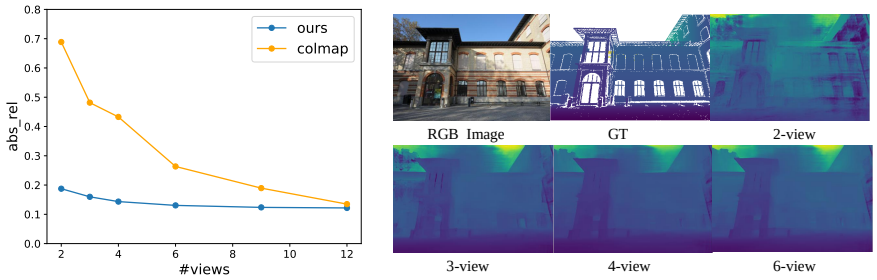


(a) depth metrics comparison (b) camera pose metrics comparison

Fig. 7. Comparison with baseline during iterations. Our work converges at a better position. (a) abs relative error and log RMSE. (b) rotation and translation error.

Iterative Improvement Our model can run iteratively to reduce the prediction error. Fig.7 (solid lines) shows our performance over iterations when initialized with the prediction from DeMoN. As can be seen, our model effectively reduces both depth and pose errors upon the DeMoN output. Throughout the iterations, better depth and pose benefit each other by building more accurate cost volume, and both are consistently improved. The whole process is similar to coordinate descent algorithm, and finally converges at iteration 4.

Effect of P-CV We compare DeepSFM to a baseline method for our P-CV. In this baseline, the depth prediction is the same as DeepSFM, but the pose prediction network is replaced by a direct visual odometry model [36], which updates camera parameters by minimizing pixel-wise photometric error between image features. Both methods are initialized with DeMoN results. As provided in Fig.7, DeepSFM consistently produces lower errors on both depth and pose over all the iterations. This shows that our P-CV predicts more accurate pose and performs more robust against noise depth at early stages. Fig. 5(c) shows



(a) Abs relative error comparison (b) Our results with different view numbers

Fig. 8. Depth map results w.r.t. the number of images. Our performance does not change much with varying number of views.

the visualized pose trajectories which are estimated by baseline(cyan) and our method(yellow) on ETH3D.

View Number DeepSFM works still reasonably well with fewer views due to the free from optimization based components. To show this, we compare to COLMAP with respect to the number of input views on ETH3D. As depicted in Fig.8, more images yield better results for both methods as expected. However, our performance drops significantly slower than COLMAP with fewer number of inputs. Numerically, DeepSFM cuts the depth error by half under the same number of views as COLMAP, or achieves similar error with half number of views required by COLMAP. This clearly demonstrates that DeepSFM is more robust when fewer inputs are available.

5 Conclusions

We present a deep learning framework for Structure-from-Motion, which explicitly enforces photo-metric consistency, geometric consistency and camera motion constraints all in the deep network. This is achieved by two key components - namely D-CV and P-CV. Both cost volumes measure the photo-metric and geometric errors by hypothetically moving reconstructed scene points (structure) or camera (motion) respectively. Our deep network can be considered as an enhanced learning based BA algorithm, which takes the best benefits from both learnable priors and geometric rules. Consequently, our method outperforms conventional BA and state-of-the-art deep learning based methods for SfM.

Acknowledgements

This project is partly supported by NSFC Projects (61702108), STCSM Projects (19511120700, and 19ZR1471800), SMSTM Project (2018SHZDZX01), SRIF Program (17DZ2260900), and ZJLab.

References

1. Agarwal, S., Furukawa, Y., Snavely, N., Simon, I., Curless, B., Seitz, S.M., Szeliski, R.: Building rome in a day. Communications of the ACM **54**(10), 105–112 (2011)
2. Agarwal, S., Snavely, N., Seitz, S.M., Szeliski, R.: Bundle adjustment in the large. In: European conference on computer vision. pp. 29–42. Springer (2010)
3. Chang, A.X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., et al.: Shapenet: An information-rich 3d model repository. arXiv preprint arXiv:1512.03012 (2015)
4. Clark, R., Bloesch, M., Czarnowski, J., Leutenegger, S., Davison, A.J.: Learning to solve nonlinear least squares for monocular stereo. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 284–299 (2018)
5. Delaunoy, A., Pollefeys, M.: Photometric bundle adjustment for dense multi-view 3d modeling. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1486–1493 (2014)
6. Eigen, D., Fergus, R.: Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In: The IEEE International Conference on Computer Vision (ICCV) (December 2015)
7. Eigen, D., Puhrsch, C., Fergus, R.: Depth map prediction from a single image using a multi-scale deep network. In: Advances in neural information processing systems. pp. 2366–2374 (2014)
8. Engel, J., Koltun, V., Cremers, D.: Direct sparse odometry. IEEE transactions on pattern analysis and machine intelligence **40**(3), 611–625 (2017)
9. Engel, J., Schöps, T., Cremers, D.: Lsd-slam: Large-scale direct monocular slam. In: European conference on computer vision. pp. 834–849. Springer (2014)
10. Fu, H., Gong, M., Wang, C., Batmanghelich, K., Tao, D.: Deep ordinal regression network for monocular depth estimation. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2018)
11. Fuhrmann, S., Langguth, F., Goesele, M.: Mve-a multi-view reconstruction environment. In: GCH. pp. 11–18 (2014)
12. Furukawa, Y., Curless, B., Seitz, S.M., Szeliski, R.: Towards internet-scale multi-view stereo. In: 2010 IEEE computer society conference on computer vision and pattern recognition. pp. 1434–1441. IEEE (2010)
13. Garg, R., BG, V.K., Carneiro, G., Reid, I.: Unsupervised cnn for single view depth estimation: Geometry to the rescue. In: European Conference on Computer Vision (ECCV). pp. 740–756. Springer (2016)
14. Gherardi, R., Farenzena, M., Fusiello, A.: Improving the efficiency of hierarchical structure-and-motion. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. pp. 1594–1600. IEEE (2010)
15. Han, X., Leung, T., Jia, Y., Sukthankar, R., Berg, A.C.: Matchnet: Unifying feature and metric learning for patch-based matching. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3279–3286 (2015)
16. Hartley, R.I.: In defense of the eight-point algorithm. IEEE Transactions on pattern analysis and machine intelligence **19**(6), 580–593 (1997)
17. Hirschmuller, H.: Accurate and efficient stereo processing by semi-global matching and mutual information. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). vol. 2, pp. 807–814. IEEE (2005)
18. Hochreiter, S., Younger, A.S., Conwell, P.R.: Learning to learn using gradient descent. In: International Conference on Artificial Neural Networks. pp. 87–94. Springer (2001)

19. Huang, P.H., Matzen, K., Kopf, J., Ahuja, N., Huang, J.B.: Deepmvs: Learning multi-view stereopsis. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2821–2830 (2018)
20. Im, S., Jeon, H.G., Lin, S., Kweon, I.S.: Dpsnet: End-to-end deep plane sweep stereo. In: International Conference on Learning Representations (2019), https://openreview.net/forum?id=ryeYHi0ctQ
21. Kaiming, H., Xiangyu, Z., Shaoqing, R., Sun, J.: Spatial pyramid pooling in deep convolutional networks for visual recognition. In: European Conference on Computer Vision (ECCV) (2014)
22. Kar, A., Häne, C., Malik, J.: Learning a multi-view stereo machine. In: Advances in neural information processing systems. pp. 365–376 (2017)
23. Knapitsch, A., Park, J., Zhou, Q.Y., Koltun, V.: Tanks and temples: Benchmarking large-scale scene reconstruction. ACM Transactions on Graphics **36**(4) (2017)
24. Kuznietsov, Y., Stuckler, J., Leibe, B.: Semi-supervised deep learning for monocular depth map prediction. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (July 2017)
25. Laina, I., Rupprecht, C., Belagiannis, V., Tombari, F., Navab, N.: Deeper depth prediction with fully convolutional residual networks. In: 2016 Fourth International Conference on 3D Vision (3DV). pp. 239–248. IEEE (2016)
26. Liu, F., Shen, C., Lin, G., Reid, I.: Learning depth from single monocular images using deep convolutional neural fields. IEEE transactions on pattern analysis and machine intelligence **38**(10), 2024–2039 (2016)
27. Lourakis, M., Argyros, A.A.: Is levenberg-marquardt the most efficient optimization algorithm for implementing bundle adjustment? In: Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1. vol. 2, pp. 1526–1531. IEEE (2005)
28. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International journal of computer vision **60**(2), 91–110 (2004)
29. Mur-Artal, R., Montiel, J.M.M., Tardos, J.D.: Orb-slam: a versatile and accurate monocular slam system. IEEE transactions on robotics **31**(5), 1147–1163 (2015)
30. Mur-Artal, R., Tardós, J.D.: Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. IEEE Transactions on Robotics **33**(5), 1255–1262 (2017)
31. Newcombe, R.A., Lovegrove, S.J., Davison, A.J.: Dtam: Dense tracking and mapping in real-time. In: 2011 international conference on computer vision. pp. 2320–2327. IEEE (2011)
32. Nocedal, J., Wright, S.: Numerical optimization. Springer Science & Business Media (2006)
33. Schonberger, J.L., Frahm, J.M.: Structure-from-motion revisited. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4104–4113 (2016)
34. Schöps, T., Schönberger, J.L., Galliani, S., Sattler, T., Schindler, K., Pollefeys, M., Geiger, A.: A multi-view stereo benchmark with high-resolution images and multi-camera videos. In: Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
35. Snavely, N.: Scene reconstruction and visualization from internet photo collections: A survey. IPSJ Transactions on Computer Vision and Applications **3**, 44–66 (2011)
36. Steinbrücker, F., Sturm, J., Cremers, D.: Real-time visual odometry from dense rgb-d images. In: 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops). pp. 719–722. IEEE (2011)

37. Sturm, J., Engelhard, N., Endres, F., Burgard, W., Cremers, D.: A benchmark for the evaluation of rgb-d slam systems. In: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems. pp. 573–580. IEEE (2012)
38. Tang, C., Tan, P.: Ba-net: Dense bundle adjustment network. arXiv preprint arXiv:1806.04807 (2018)
39. Teed, Z., Deng, J.: Deepv2d: Video to depth with differentiable structure from motion. arXiv preprint arXiv:1812.04605 (2018)
40. Triggs, B., McLauchlan, P.F., Hartley, R.I., Fitzgibbon, A.W.: Bundle adjustment—a modern synthesis. In: International workshop on vision algorithms. pp. 298–372. Springer (1999)
41. Ummenhofer, B., Zhou, H., Uhrig, J., Mayer, N., Ilg, E., Dosovitskiy, A., Brox, T.: Demon: Depth and motion network for learning monocular stereo. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5038–5047 (2017)
42. Vijayanarasimhan, S., Ricco, S., Schmid, C., Sukthankar, R., Fragkiadaki, K.: Sfm-net: Learning of structure and motion from video. arXiv preprint arXiv:1704.07804 (2017)
43. Wang, C., Miguel Buenaposada, J., Zhu, R., Lucey, S.: Learning depth from monocular videos using direct methods. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2022–2030 (2018)
44. Wang, P., Shen, X., Lin, Z., Cohen, S., Price, B., Yuille, A.L.: Towards unified depth and semantic prediction from a single image. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2015)
45. Wang, S., Clark, R., Wen, H., Trigoni, N.: Deepvo: Towards end-to-end visual odometry with deep recurrent convolutional neural networks. In: 2017 IEEE International Conference on Robotics and Automation (ICRA). pp. 2043–2050. IEEE (2017)
46. Wu, C., Agarwal, S., Curless, B., Seitz, S.M.: Multicore bundle adjustment. In: CVPR 2011. pp. 3057–3064. IEEE (2011)
47. Wu, C., et al.: Visualsfm: A visual structure from motion system, 2011. URL http://www. cs. washington. edu/homes/ccwu/vsfm **14**, 2 (2011)
48. Xiao, J., Owens, A., Torralba, A.: Sun3d: A database of big spaces reconstructed using sfm and object labels. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1625–1632 (2013)
49. Xu, Q., Tao, W.: Multi-scale geometric consistency guided multi-view stereo. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5483–5492 (2019)
50. Yao, Y., Luo, Z., Li, S., Fang, T., Quan, L.: Mvsnet: Depth inference for unstructured multi-view stereo. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 767–783 (2018)
51. Yao, Y., Luo, Z., Li, S., Shen, T., Fang, T., Quan, L.: Recurrent mvsnet for high-resolution multi-view stereo depth inference. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5525–5534 (2019)
52. Zhou, T., Brown, M., Snavely, N., Lowe, D.G.: Unsupervised learning of depth and ego-motion from video. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1851–1858 (2017)