Multi-Scale Geometric Consistency Guided Multi-View Stereo

Qingshan Xu and Wenbing Tao*
National Key Laboratory of Science and Technology on Multispectral Information Processing
School of Artifical Intelligence and Automation, Huazhong University of Science and Technology, China
{qingshanxu, wenbingtao}@hust.edu.cn

Abstract

In this paper, we propose an efficient multi-scale geometric consistency guided multi-view stereo method for accurate and complete depth map estimation. We first present our basic multi-view stereo method with Adaptive Checkerboard sampling and Multi-Hypothesis joint view selection (ACMH). It leverages structured region information to sample better candidate hypotheses for propagation and infer the aggregation view subset at each pixel. For the depth estimation of low-textured areas, we further propose to combine ACMH with multi-scale geometric consistency guidance (ACMM) to obtain the reliable depth estimates for low-textured areas at coarser scales and guarantee that they can be propagated to finer scales. To correct the erroneous estimates propagated from the coarser scales, we present a novel detail restorer. Experiments on extensive datasets show our method achieves state-of-the-art performance, recovering the depth estimation not only in low-textured areas but also in details.

1. Introduction

Multi-view stereo (MVS) has traditionally been a topic of interest in computer vision for decades. It aims at establishing dense correspondence from multiple calibrated images, which results in a dense 3D reconstruction. Over the last few years, much effort has been put into improving the quality of dense 3D reconstructions and some works have achieved impressive results [7, 8, 9, 23, 24, 25, 36, 19]. However, with the large-scale data, low texture, occlusions, repetitive patterns and reflective surface, it is still a challenging problem to perform efficient and accurate multi-view stereo in computer vision domain.

Recently, PatchMatch Stereo methods [1, 36, 8, 19] show great power in depth map estimation with their fast global search for the best match in other images [2]. These methods follow a popular four-step pipeline, including random initialization, propagation, view selection and refinement. In this pipeline, propagation and view selection are two key steps to PatchMatch Stereo methods. The former is important to efficiency while the latter is critical to accuracy.

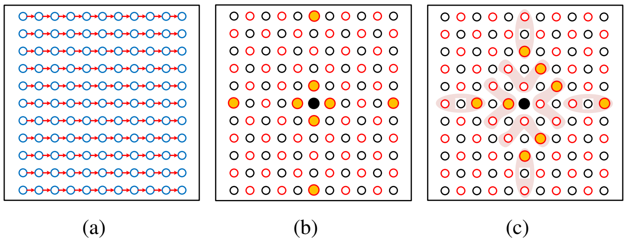


Figure 1. Propagation scheme. (a) Sequential propagation. (b) Symmetric checkerboard propagation. (c) Adaptive checkerboard propagation. The light red areas in (c) show sampling regions. The solid yellow circles in (b) and (c) show the sampled points.

For propagation, there generally exist two distinct types of parallel schemes: sequential propagation [1, 36, 19] and diffusion-like propagation [8]. The former traverses pixels following parallel scanlines only in the vertical (or horizontal) direction (Figure 1(a)). In contrast, the later simultaneously updates the status of half of the pixels in an image with a checkerboard pattern (Figure 1(b)). In terms of efficiency, the diffusion-like propagation achieves better algorithm parallelism. However, it is reported in [19, 21] that, its reconstruction results are not competitive with the sequential propagation's in some challenging cases. As pointed out in [36], this mainly attributes to its less robust view selection instead of propagation. For example, in the sequential propagation, [36, 19] construct a probabilistic graphical model to perform pixelwise view selection. Unlike their elaborate view selection, the diffusion-like propagation adopts a simple threshold truncation scheme to determine aggregation view subsets [8]. This leads to its biased view selection for different hypotheses. Then a motivating question is, whether it is possible to design a more robust view selection based on the checkerboard pattern.

To this end, we first propose our basic MVS method with Adaptive Checkerboard sampling and Multi-Hypothesis

[*] Corresponding author

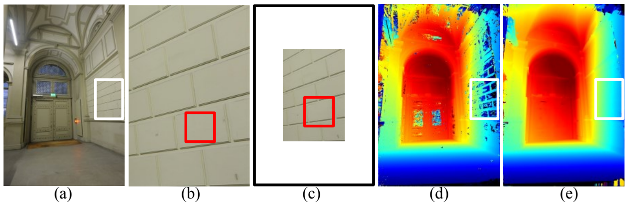


Figure 2. Texture richness for different scales. (a) Original Image. (b) The zoomed-in version of the white box in (a). (c) The downsampled version of (b). (d) Depth map obtained with the original scale. (e) Depth map obtained with the multi-scale scheme. The patch windows in red are kept the same size.

joint view selection (ACMH). Our key idea is based on the assumption of [18] that pixels within a relatively large region can be approximately modeled by one 3D plane, which indicates *structured region information* and a shared hypothesis among these pixels. Thus, unlike fixed sampling in diffusion-based conventions which may be misleading, ACMH searches larger regions to adaptively sample better candidate hypotheses for propagation (Figure 1(c)). With these better hypotheses, we propose a multi-hypothesis joint strategy to infer pixelwise view selection. For a specific pixel, this strategy employs a voting scheme to supply the same aggregation view subset for different propagated hypotheses, and gives credible views greater weights to aggregate the final multi-view matching cost. As a result, ACMH can achieve accurate depth map estimation while inheriting the high efficiency of the checkerboard pattern.

Moreover, as a key component of PatchMatch Stereo methods, view selection heavily depends on a stable visual similarity measure between two image patches. However, measuring the visual similarity in low-textured areas is always challenging. As depicted in Figure 2(b), the low discrimination in low-textured areas leads to the ambiguity of visual similarity, which further degrades the performance of PatchMatch Stereo methods (Figure 2(d)). However, we observe that, for the low-textured areas, though the texture information with an universal patch window in Figure 2(b) is not significant , it becomes more discriminative under the same patch window when an image is downsampled (Figure 2(c)). That is, the texture richness is a relative measure. Then, an intuitive idea is that, we can estimate depth information at coarser scales to alleviate the ambiguities in low-textured areas and use it as guidance for the matching progress at finer scales.

Based on the above idea, we further present a multi-scale patch matching with geometric consistency guidance, called ACMM. Specifically, our method constructs image pyramids and obtains reliable depth estimates for low-textured areas at coarser scales. After propagating these estimates from coarser scales to finer scales via upsampling, we resort to geometric consistency to constrain the depth optimization at finer scales. Considering that the depth propagation from coarser scales to finer scales often leads to depth information loss in details, we present a detail restorer based on the difference map of photometric consistency between adjacent scales. Through our proposed strategies, our approach can not only estimate depth information in low-textured areas but also preserve details.

Our main contributions are summarized as follows: 1) Inherited from the high efficiency of the diffusion-like propagation, we present an adaptive checkerboard sampling scheme to select more reasonable hypotheses for propagation based on the structured region information. Then, a multi-hypothesis joint view selection is proposed to help select credible aggregation views. 2) For the ambiguities in low-textured areas, we propose a multi-scale patch matching scheme with geometric consistency guidance. The geometric consistency imposed at different scales can guarantee that the reliable depth estimates for low-textured areas obtained at coarser scales are retained at finer scales. Moreover, a detail restorer is present to correct errors propagated from the coarser scales. Through extensive evaluation, we demonstrate the effectiveness and efficiency of our method by achieving state-of-the-art performance on Strecha dataset [27] and ETH3D benchmark[21].

2. Related Work

According to [22], MVS methods can be categorized into four groups, voxel-based methods [5, 29, 26], surface evolution based methods [4, 11, 3], patch-based methods [9, 17, 7] and depth map based methods [36, 8, 19]. The voxel-based methods are often constrained by their predefined voxel grid resolution. The surface evolution based methods depend on a good initial solution. As for the patch-based methods, its dependence on matched keypoints impairs the completeness of 3D models. The depth map based methods require estimating depth maps for all images and then fusing them into a unified 3D scene representation. A more detailed overview of MVS methods is presented in [22, 6]. Our method belongs to the last category and we only discuss the related PatchMatch Stereo approaches.

In terms of efficiency, [1, 30, 36, 19] adopt the sequential propagation scheme. They alternatively perform upward/downward propagation in odd iteration steps and perform leftward/rightward propagation in even steps. To increase parallelism, [30] selects an eighth of the image height (width) as the length of each scanline in the vertical (horizontal) propagation. However, the algorithm parallelism of sequential propagation is still proportional to the number of rows or columns of images. Then, Galliani *et al.* [8] propose to leverage a checkerboard pattern to perform a diffusion-like propagation scheme. It allows to simultaneously update the status of half of the pixels in an image. However, they ignore good hypotheses should have priority in propagation.

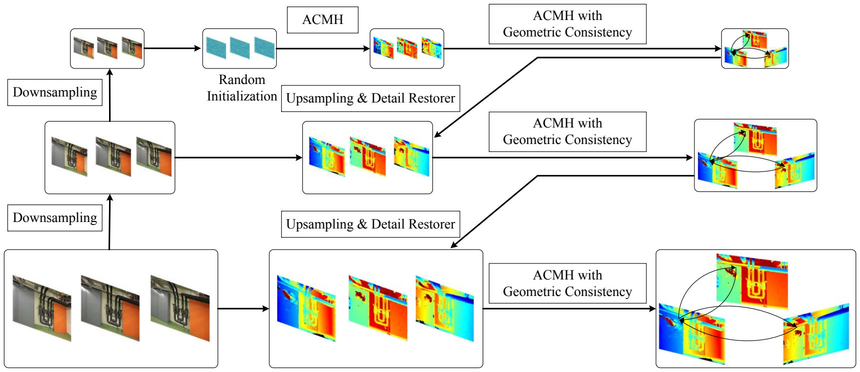

ACMH
ACMH with Geometric Consistency
Downsampling
Random Initialization
Upsampling & Detail Restorer
ACMH with Geometric Consistency
Downsampling
Upsampling & Detail Restorer
ACMH with Geometric Consistency

Figure 3. Overview of our approach. The initial depth maps of the coarsest scale are obtained by our basic MVS model with only photometric consistency (Section 4). After upsampling the estimation of the previous scale to the current scale, detail restorer is implemented to correct the errors in details. At each scale, geometric consistency is enforced to enhance coherence and prevent the reliable estimates in low-textured areas from the previous scale being impaired by photometric consistency (Section 5).

According to the above propagation strategies, many view selection schemes are proposed to tackle the noise in the propagation process. In the diffusion-like propagation scheme, [8] selects fixed k views with the minimal k matching costs. However, this leads to a bias due to different aggregation subsets for different hypotheses. In the sequential propagation, [1, 30] also ignore the pixelwise view selection by only demanding global view angles. To incorporate only useful neighboring views at each pixel, Zheng *et al*. [36] first try to construct a probabilistic graphical model to jointly estimate depth maps and view selection. Further, Schönberger *et al*. [19] introduce geometric priors and temporal smoothness to better depict the state-transition probability. However, this sequential inference needs to condition the status of previous pixels at the current state. It is still more sensitive to noise in low-textured areas.

Although some methods focus on view selection to improve local smoothness and gain some benefits, they are still restricted by patch window size. To perceive more useful information in low-textured areas, Wei *et al*. [30] adopt the multi-scale patch matching with variance based consistency. However, this consistency is too strong to spread some reliable estimates in few neighboring views across multiple views. Moreover, they overlook the errors in details.

3. Overview

Given a set of input images $\mathcal{I} = \{I_i \,|\, i = 1 \cdots N\}$ with known calibrated camera parameters $\mathcal{P} = \{P_i \,|\, i = 1 \cdots N\}$, our goal is to estimate depth maps $\mathcal{D} = \{D_i \,|\, i = 1 \cdots N\}$ for all images and fuse them into a 3D point cloud. Specifically, we aim to recover the depth map for reference image I_{ref} sequentially selected from $\mathcal{I}$ with the guidance of source images I_{src} $(\mathcal{I} - I_{\text{ref}})$.

An overview of our method is illustrated in Figure 3. We construct a pyramid with k scales for all images with a downsampling factor η. We denote the l-th scale of I_i and corresponding camera parameter as I_i^l and P_i^l, $l = 0 \cdots k-1$. The finest scale of the pyramids I_i^{k-1} are the raw images. We aim to propagate the reliable estimates in low-textured areas from coarser scales to help with the estimation of finer scales without much loss in details.

We first use our basic MVS model with photometric consistency, ACMH, to obtain the initial depth maps for all images at the coarsest scale. To enhance the coherence among all depth maps, we further perform ACMH with geometric consistency. Then we upsample the depth maps to the next scale. The upsampling propagates the reliable depth estimates in low-textured areas to the current scale, which are obtained at the previous scale. To correct the errors induced from the previous scale, a detail restorer is first employed. These corrected depth maps are utilized as initialization to guide the subsequent ACMH with geometric consistency such that the reliable estimates within low-textured areas can be kept and optimized at the current scale. The same upsampling, detail restorer and ACMH with geometric consistency are repeated until we obtain the depth maps at the original image scale. We term our whole method ACMM.

4. Structured Region Information

Structured region information means that pixels within a relatively large region can be approximately be modeled by the same 3D plane. Our basic MVS method with Adaptive Checkerboard sampling and Multi-Hypothesis joint view selection (ACMH) is inspired by this to sample better candidate hypotheses for propagation and select views with more credibility for multi-view matching costs aggregation. The details of ACMH are given as follows.

4.1. Random Initialization

Following [8], we first randomly generate a hypothesis (including depth and normal) to build a 3D plane for each pixel in the reference image I_{ref}. For each hypothesis, a matching cost is computed from each of $N-1$ source images via a plane-induced homography [10]. Then the top K best matching costs are aggregated into the initial multi-view matching cost for the subsequent propagation.

4.2. Adaptive Checkerboard Sampling

We first adopt the idea in [8] to partition the pixels of I_{ref} into red-black grids of a checkerboard. This pattern allows us to simultaneously update the hypotheses of black pixels using red pixels and vice versa. In [8], their method samples from eight fixed positions. Differently, for each pixel in red or black group, we expand these eight points into four V-shaped areas and four long strip areas (Figure 1(c)). Each V-shaped area contains 7 samples while every long strip area contains 11 samples. Then we sample eight good hypotheses from these areas according to their previous multi-view matching costs. This sampling scheme is favored by the structured region information. It means that a hypothesis with a smaller multi-view matching cost will represent a local plane better. This strategy helps a good plane of a local shared region to spread further as much as possible and supplies more compact estimates.

4.3. Multi-Hypothesis Joint View Selection

To obtain a robust multi-view matching cost for each pixel, we further leverage these eight structured hypotheses to infer the weight of every neighboring views. For pixel p, we calculate its corresponding matching costs with propagated hypotheses and embed them into a cost matrix

$$\mathsf{M} = \begin{bmatrix} m_{1,1} & m_{1,2} & \cdots & m_{1,N-1} \\ m_{2,1} & m_{2,2} & \cdots & m_{2,N-1} \\ \vdots & \vdots & \ddots & \vdots \\ m_{8,1} & m_{8,2} & \cdots & m_{8,N-1} \end{bmatrix}, \tag{1}$$

where $m_{i,j}$ is the matching cost for the i-th hypothesis h_i scored by the j-th view I_j. We adopt the bilateral weighted adaption of normalized cross correlation [19] to compute the matching cost, which describes the *photometric consistency* between the reference and source patch.

To infer aggregation views from the above cost matrix, we apply a voting decision in each column to determine whether a view is appropriate. A key observation behind this is that for a bad view, its corresponding eight matching costs are always high. In contrast, a good view always has some smaller matching costs. Furthermore, the matching costs for the good view will decrease with the iteration of our algorithm. Therefore, a good matching cost boundary is defined as

$$\tau(t) = \tau_0 \cdot e^{-\frac{t^2}{\alpha}}, \tag{2}$$

where t means the t-th iteration, τ_0 is the initial matching cost threshold and α is a constant. Besides, we define a fixed bad matching cost threshold τ_1 ($\tau_1 > \tau(t)$). Based on our above observation, for a specific view I_j, there should exist more than n_1 matching costs meeting the condition: $m_{i,j} < \tau(t)$. We define this set as $\mathrm{S}_{\text{good}}(j)$ to calculate the weight of view I_j later. Also, there should be less than n_2 matching costs meeting the condition: $m_{i,j} > \tau_1$. A view simultaneously satisfying the above conditions will be incorporated into the current view selection set S_t in the t-th iteration.

The above inferred view selection set S_t may contain some unstable views because of noise, viewing point and scale, *etc*. This means each selected view will contribute different weights to the final aggregated matching cost. To evaluate the importance of each selected view, the confidence of a matching cost is computed as follows,

$$C(m_{ij}) = e^{-\frac{m_{ij}^2}{2\beta^2}}. \tag{3}$$

where β is a constant. This makes good views more discriminative. The weight of each selected view can be defined as

$$w(I_j) = \frac{1}{|\mathrm{S}_{\text{good}}(j)|} \sum_{m_{i,j} \in \mathrm{S}_{\text{good}}(j)} C(m_{i,j}), I_j \in \mathrm{S}_t. \tag{4}$$

We suppose the most important view v_{t-1} in iteration $t-1$ shall continue to have influence on the view selection of the current iteration t. Thus, we modify Formula 4 as

$$w'(I_j) = \begin{cases} (\mathbb{I}(I_j = v_{t-1}) + 1) \cdot w(I_j), & \text{if } I_j \in \mathrm{S}_t; \\ 0.2 \cdot \mathbb{I}(I_j = v_{t-1}), & \text{else.} \end{cases} \tag{5}$$

where $\mathbb{I}(\cdot)$ is an indicator function such that $\mathbb{I}(\text{true}) = 1$ and $\mathbb{I}(\text{false}) = 0$. This modification can make our view selection method more robust. With the inferred weights w', the multi-view aggregated photometric consistency cost of pixel p for hypothesis h_i is defined as

$$m_{\text{photo}}(p, h_i) = \frac{\sum_{j=1}^{N-1} w'(I_j) \cdot m_{i,j}}{\sum_{j=1}^{N-1} w'(I_j)}. \tag{6}$$

The current best estimate for pixel p is updated by the hypothesis with the minimum multi-view aggregated cost.

4.4. Refinement

After each red-black iteration, a refinement step is applied to enrich the diversity of solution space. There exist three conditions for the current depth and normal of pixel p, *i.e.*, either of them, neither of them, or both of them are close to the optimal solution [19]. Thus, we generate two new hypotheses, one of which is randomly generated and the other is obtained by perturbing the current estimate. We combine these new depths and normals with the current depth and normal, yielding another six new hypotheses to be tested. The hypothesis with the least aggregated cost is chosen as the final estimate for pixel p. The above propagation, view selection and refinement are repeated multiple times to get the final depth map for I_{ref}. At the end, a median filter of size 5 × 5 is applied to our final depth maps.

5. Multi-Scale Geometric Consistency

Combined with the multi-scale scheme, ACMH at the coarsest scale obtains more reliable depth estimates in low-textured areas. However, photometric consistency experiences difficulties when applied to optimize these depth estimates at finer scales. In this section, we detail how to leverage geometric consistency guidance to deal with the optimization of these estimates. Also, a detail restorer is present to correct the errors induced from coarser scales.

5.1. Geometric Consistency Guidance

After obtaining the reliable depth estimates for low-texture areas at the coarsest scale and propagating them to finer scales via upsampling, we need to optimize these estimates at finer scales. Our key idea is that the upsampled depth maps of source images can geometrically constrain these estimates from being disturbed by photometric consistency, which means *geometric consistency*. Inspired by [34, 19], we use the forward-backward reprojection error to indicate this consistency.

Given the depth of pixel p in image I_i is known as $D_i(p)$, with the camera parameter $P_i = [M_i|\mathrm{p}_{i,4}]$ [10], its corresponding back-projected 3D point $X_i(p)$ is computed as

$$X_i(p) = M_i^{-1} \cdot (D_i(p) \cdot p - \mathrm{p}_{i,4}). \tag{7}$$

Then the reprojection error between the reference image I_{ref} and the source image I_j for i-th hypothesis is calculated as

$$\Delta e_{i,j} = \min(\|P_{\text{ref}} \cdot X_j(P_j \cdot X_{\text{ref}}(p)) - p\|, \delta), \tag{8}$$

where δ is a truncation threshold to robustify the reprojection error against occlusions. We integrate the above equation into Formula 6 and get the following multi-view aggregated geometric consistency cost as

$$m_{\text{geo}}(p, h_i) = \frac{\sum_{j=1}^{N-1} w'(I_j) \cdot (m_{i,j} + \lambda \cdot \Delta e_{i,j})}{\sum_{j=1}^{N-1} w'(I_j)}, \tag{9}$$

where λ is a factor that balances the weight of the two terms.

Specifically, at the l-th scale ($l > 0$), we employ the joint bilateral upsampler [15] to propagate the estimates at the previous scale to the current scale. The upsampled estimates are utilized as the initial seeds of the current scale to perform the subsequent propagation, view selection and refinement as in ACMH. Differently, here we adopt Formula 9 instead of Formula 6 to update the hypothesis of pixel p. In fact, this modification limits the solution space of current hypothesis update, especially for the hypothesis update in low-textured areas. This guarantees that the reliable estimates in low-textured areas obtained at the coarsest scale can be propagated to the finest scale. It is worth noting that the geometric consistency also optimizes the depth estimation of other areas except low-textured areas.

Additionally, we notice that the initial depth maps obtained by ACMH are noisy due to ambiguities and occlusions. However, photometric consistency is hard to reflect these errors since large depth variations only induce small cost changes [19]. Thus, we also perform geometric consistency at the coarsest scale to optimize these initial depth maps. Intuitively, if the neighboring depth maps are estimated more accurately, the depth map of the reference image will be further boosted. Thus, we conduct geometric consistency guidance twice to refine depth maps at each scale in our experiments.

5.2. Detail Restorer

The multi-scale geometric consistency guidance on the one hand helps with the estimation of low-textured areas but on the other hand often leads to blurred details. At the coarser scales, the lost image details directly cause the loss of their depth information. Additionally, the fixed patch window size makes ACMH hard to achieve a trade-off between thin structures and low-textured areas because the local planar assumption does not hold for details [14, 35, 31]. Furthermore, although upsampling can spread the reliable estimates in low-textured areas to larger regions, it also brings some extra errors in details. However, we observe that these details can be better estimated at the original image scale with only photometric consistency (Figure 4(c)). Thus, we consider how to leverage photometric consistency to probe erroneous estimates in details and correct them.

As shown in Figure 4(a), the blurred details often happen in thin structures or boundaries. We hope to detect these regions and only enforce photometric consistency in these specific regions to rectify the erroneous estimates. We observe that the difference map of photometric consistency cost between adjacent scales can magnify the errors in details while suppressing the reflection of reliable estimates in low-textured areas (Figure 4(e)). Thus, we can leverage this difference map to probe the errors in details and correct them in a unified way. Specifically, after we upsample the estimates (*i.e.*, depth and normal) of the previous scale, we use them to recompute the photometric consistency cost C_{init}^l at the current scale. Then, we execute the basic MVS model to get new photometric consistency cost C_{photo}^l. The estimate for pixel p will be considered as an error if the difference of photometric consistency cost fulfills

$$C_{\text{init}}^l(p) - C_{\text{photo}}^l(p) > \xi, \tag{10}$$

where ξ is a small constant value that increases the robustness to distinguish the erroneous estimates. Meanwhile, the erroneous estimates will be replaced by the hypotheses reflecting the above difference. By combining the detail restorer, ACMM can make a better trade-off between low-textured areas and details as shown in Figure 4(f).

6. Fusion

After getting all depth maps, we adopt a fusion step similar to [8, 19] to merge them into a complete point cloud.

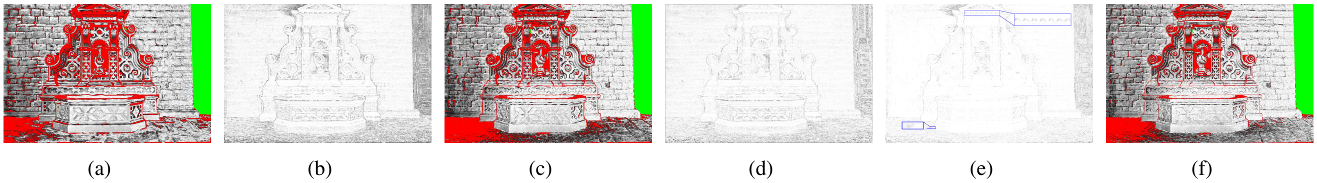


Figure 4. Absolute error maps and photometric consistency cost maps on Fountain-P11 dataset for different methods. (a) shows the absolute error map of our method without detail restorer. Its depth map is obtained by upscaling the estimation of the penultimate scale. Details are not preserved. (b) shows the photometric consistency cost map of (a). (c) shows the absolute error map of our basic MVS model. Its details are better preserved than (a). (d) shows the photometric consistency cost map of (c). (e) shows the difference map of (b) and (d). The cost difference of the erroneous estimates in details is more discriminative than the cost difference of the reliable estimation in low-textured areas. (f) shows the absolute error map of ACMM. For absolute error maps, green pixels encode missing ground truth data, red pixels encode an absolute error larger than $2cm$, and pixels with absolute errors between 0 and $2cm$ are encoded in gray [255, 0].

Specifically, we cast each image as reference image in turn, convert its depth map to 3D points in world coordinate and project them to its neighboring views to get corresponding matches. We define a consistent match satisfying the relative depth difference $\epsilon \leq 0.01$, the angle between normals $\theta \leq 30°$ and the reprojection error $\psi \leq 2$ as in [19]. If there exist $n \geq 2$ neighboring views whose corresponding matches satisfy the above constraints, the depth estimate will be accept. At last, the 3D points and normal estimates corresponding to these consistent depth estimates are averaged into a unified 3D point.

7. Experiments

We evaluate our method on two MVS datasets, Strecha dataset [27] and ETH3D benchmark [21], from two perspectives, depth map assessment and point cloud evaluation.

7.1. Datasets and Settings

Strecha dataset [27] comprises two scenes with ground truth depth maps, Fountain and HerzJesu. They have 11 and 8 images respectively with 3072×2048 resolution. Although Strecha dataset provides relatively easy (*i.e.*, well-textured) scenes and its online service is not available anymore, there are many state-of-the-art methods evaluating their depth maps on it. Thus, we will first utilize Strecha dataset to assess the quality of depth maps. ETH3D benchmark [21] consists of three scenarios corresponding to different tasks for (multi-view) stereo algorithms. It is more challenging for containing a diverse set of viewpoints and scene types. Here we only focus on high-resolution multi-view stereo dataset with images at a resolution of 6048×4032[1]. Additionally, the high-resolution multi-view stereo dataset contains training datasets and test datasets. The training datasets provide not only ground truth point clouds but also ground truth depth maps, while the ground truth of the test datasets is withheld by the benchmark's web site.

All of our experiments are conducted on a machine with two Intel E5-2630 CPUs and two GTX Titan X GPUs. In the multi-hypothesis joint view selection scheme, $\{\tau_0, \tau_1, \alpha, \beta, n_1, n_2\} = \{0.8, 1.2, 90, 0.3, 2, 3\}$. In our geometric consistency guidance strategy and detail restorer, $\{k, \eta, \delta, \lambda, \xi\} = \{3, 0.5, 3, 0.2, 0.1\}$. Note that, we use only every other row and column in the window to speed up the computation of matching cost [8].

7.2. Depth Map Evaluation

We evaluate our method's effectiveness on depth map estimation on Strecha dataset and ETH3D benchmark in this section. Following [12], we calculate the percentage of pixels with a absolute depth error less than $2cm$ and $10cm$ from the ground truth in Table 1. To show the effectiveness of the structured region information, we replace the adaptive checkerboard sampling and multi-hypothesis joint view selection in ACMH with the diffusion-like propagation and top-k-winners-take-all view selection, denoted as DWTA.

As can be seen, with the structured region information, ACMH performs better than DWTA and is also competitive with COLMAP [19] without geometric consistency. Furthermore, we see that ACMM surpasses ACMH by a noteworthy margin and almost achieves the best performance in this dataset. Note that, HerzJesu contains more low-textured areas and CMPMVS[13] performs a bit better than ACMM on it in the case of $2cm$. This is because CMPMVS is a global energy-based method that mainly focuses on weakly-supported surface. However, on the Fountain dataset that contains more details, ACMM is much better than CMPMVS in the case of $2cm$. We also note that, COLMAP is a representative algorithm among local methods. ACMM outperforms COLMAP in the case of $2cm$, although there is no significant difference in the case of $10cm$.

To reflect more challenges such as low-textured areas and thin structures in real-world scenes, we further compare our reconstructed depth maps with COLMAP[2] on the high-

[1] In fact, we resize this imagery to no more than 3200 pixels for each dimension as [19] does.

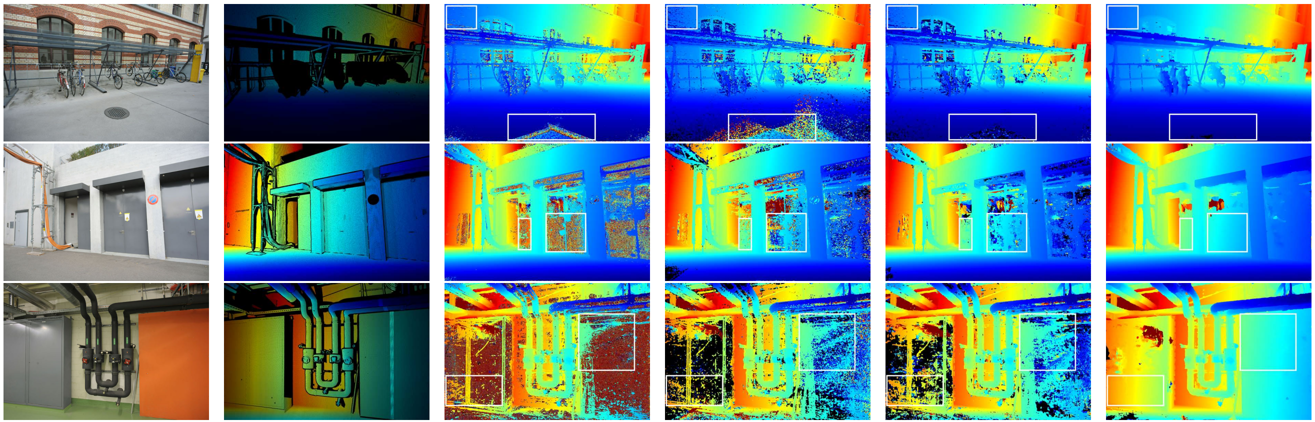
[2] Note that, the depth maps of COLMAP are obtained with its default parameters and are unfiltered.

Table 1. Percentage of pixels with absolute errors below $2cm$ and $10cm$ on Strecha dataset. The related values are from [12, 36, 19]. [19]$\backslash$G means COLMAP without geometric consistency.

	error	[36]	[32]	[7]	[33]	[28]	[13]	[8]	[19]	[19]$\backslash$G	DWTA	ACMH	ACMM
Fountain	$2cm$	0.769	0.754	0.731	0.712	0.732	0.824	0.693	0.827	0.804	0.778	0.793	**0.853**
	$10cm$	0.929	0.930	0.838	0.832	0.822	0.973	0.838	**0.975**	0.949	0.921	0.952	0.974
HerzJesu	$2cm$	0.650	0.649	0.646	0.220	0.658	**0.739**	0.283	0.691	0.679	0.614	0.656	0.731
	$10cm$	0.844	0.848	0.836	0.501	0.852	0.923	0.455	0.931	0.907	0.804	0.873	**0.932**

Table 2. Percentage of pixels with absolute errors below $2cm$ and $10cm$ on the high-resolution multi-view training datasets of ETH3D benchmark. The best results are marked in bold while the second-best results are marked in red.

error	method	indoor							outdoor					
		delive.	kicker	office	pipes	relief	relief.	terrai.	courty.	electro	facade	meadow	playgr.	terrace
$2cm$	[19]	0.697	0.435	0.263	0.411	0.863	0.858	0.576	0.826	0.710	0.742	0.546	0.709	0.808
	DWTA	0.705	0.369	0.293	0.419	0.887	0.883	0.675	0.772	0.730	0.684	0.464	0.731	0.801
	ACMH	0.733	0.427	0.323	0.536	0.891	0.903	0.714	0.799	0.748	0.685	0.571	0.753	0.820
	ACMM	**0.777**	**0.667**	**0.512**	**0.765**	**0.960**	**0.957**	**0.854**	**0.844**	**0.868**	**0.745**	**0.771**	**0.843**	**0.897**
$10cm$	[19]	0.806	0.514	0.342	0.478	0.896	0.893	0.635	0.934	0.774	0.909	0.701	0.810	0.891
	DWTA	0.815	0.451	0.382	0.496	0.918	0.918	0.738	0.910	0.810	0.899	0.647	0.844	0.894
	ACMH	0.842	0.519	0.418	0.617	0.923	0.941	0.778	0.937	0.834	0.908	0.786	0.869	0.915
	ACMM	**0.930**	**0.800**	**0.648**	**0.839**	**0.982**	**0.984**	**0.904**	**0.973**	**0.947**	**0.934**	**0.917**	**0.951**	**0.980**



(a) Reference image (b) Ground truth (c) COLMAP (d) DWTA (e) ACMH (f) ACMM

Figure 5. Qualitative depth map comparisons between different algorithms on some high-resolution multi-view training datasets (courty., electro, pipes) of ETH3D benchmark. Black pixels in (b) have no ground truth data. Some challenging areas are shown in white boxes.

resolution multi-view training datasets of ETH3D benchmark in Table 2. We see that ACMM clearly outperforms COLMAP in these challenging datasets, especially in some indoor datasets including poorly textured regions, such as kicker, office and pipes. Moreover, ACMH almost achieves the second-best performance. Figure 5 illustrates some examples of the depth maps estimated by COLMAP, DWTA, ACMH and ACMM. As can be seen, ACMH also performs better than DWTA and itself yields more robust results than COLMAP and DWTA in low-textured areas as it leverages the structured region information. Note that, although COLMAP outperforms DWTA and ACMH in some well-textured datasets such as court. and facade, it performs worse than DWTA and ACMH in some challenging datasets such as electro and office. This is because COLMAP cannot gain robust belief in challenging regions to infer pixelwise view selection. Combined with the multi-scale scheme, ACMM can further boost the estimation in these regions. Moreover, the details are also kept.

7.3. Point Cloud Evaluation

In this section, fusion is imposed to get more consistent point clouds. We evaluate our point clouds on the high-resolution multi-view test datasets of ETH3D benchmark.

Table 3 lists the accuracy, completeness and F_1 score of the point clouds estimated by PMVS [7], Gipuma [8], LTVRE [16], COLMAP, ACMH and ACMM. All these methods show similar results in accuracy. In terms of F_1 score, ACMH is competitive with other methods for its good depth map estimation. And, ACMM outperforms other methods as it inherits the structured region property of ACMH and combines with the multi-scale scheme. Furthermore, ACMM obtains much higher completeness than other methods on indoor datasets that contain more low-textured areas. This is because ACMM perceives more credible information in these areas. As for outdoor datasets,

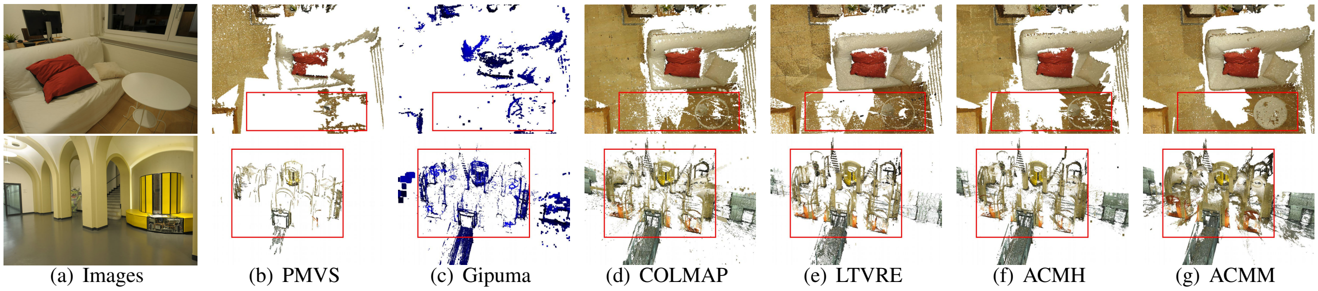

(a) Images (b) PMVS (c) Gipuma (d) COLMAP (e) LTVRE (f) ACMH (g) ACMM

Figure 6. Qualitative point cloud comparisons between different algorithms on some high-resolution multi-view test datasets (living., old co.) of ETH3D benchmark. These dense 3D models are reported by the ETH3D benchmark evaluation server [20].

Table 3. Point cloud evaluation on the high-resolution multi-view test datasets of ETH3D benchmark showing accuracy / completeness / F_1 score (in %) at different evaluation thresholds (including $2cm$ and $10cm$). The related values are from [20].

	method	$2cm$	$10cm$
indoor	PMVS	90.66 / 28.16 / 40.28	96.97 / 42.50 / 55.40
	Gipuma	86.33 / 31.44 / 41.86	98.31 / 52.22 / 65.41
	LTVRE	**93.44** / 63.54 / 74.54	**99.34** / 82.72 / 89.92
	COLMAP	91.95 / 59.65 / 70.41	98.11 / 82.82 / 89.28
	ACMH	91.14 / 64.81 / 73.93	98.76 / 82.61 / 89.42
	ACMM	90.99 / **72.73** / **79.84**	97.79 / **88.22** / **92.50**
outdoor	PMVS	88.34 / 42.89 / 55.82	95.95 / 55.17 / 68.12
	Gipuma	78.78 / 45.30 / 55.16	97.36 / 62.40 / 75.18
	LTVRE	91.82 / 74.45 / 81.41	98.72 / 90.18 / 94.19
	COLMAP	**92.04** / 72.98 / 80.81	98.64 / 89.70 / 93.79
	ACMH	83.96 / **80.03** / 81.77	97.51 / **90.57** / 93.87
	ACMM	89.63 / 79.17 / **83.58**	**98.85** / 90.43 / **94.35**
all	PMVS	90.08 / 31.84 / 44.16	96.71 / 45.67 / 58.58
	Gipuma	84.44 / 34.91 / 45.18	98.07 / 54.77 / 67.86
	LTVRE	**93.04** / 66.27 / 76.25	**99.18** / 84.59 / 90.99
	COLMAP	91.97 / 62.98 / 73.01	98.25 / 84.54 / 90.40
	ACMH	89.34 / 68.62 / 75.89	98.44 / 84.60 / 90.53
	ACMM	90.65 / **74.34** / **80.78**	98.05 / **88.77** / **92.96**

ACMM achieves almost the same completeness as ACMH does. Figure 6 illustrates some qualitative results achieved by these methods. It can be observed that, ACMM produces more complete point clouds especially in the challenging areas, *e.g.*, red boxes shown in Figure 6.

7.4. Runtime Performance

We list the runtime of depth map generation for different methods that belong to the scope of PatchMatch Stereo in Table 4. All these methods are conducted on a single GPU through our same platform[3]. ACMH and Gipuma both converge after 6 iterations while COLMAP adopts 10 iterations. For ACMM, it needs 7 iterations at the coarsest scale and 6 iterations at other scales. As Table 4 shows, ACMH is around 6× faster than COLMAP. This is because the sequential propagation of COLMAP only updates the status of one row (column) of pixels at a time and its each iteration needs propagations in 4 directions. Though ACMH and Gipuma both leverage the checkerboard propagation, ACMH is also faster than Gipuma. This is mainly because Gipuma employs a bisection refinement, which produces more unnecessary hypotheses to test. As for ACMM, it spends extra computational time on multi-scale geometric consistency scheme. However, ACMM takes no more than twice the runtime spent by ACMH as its geometric consistency at the coarser scales is conducted on downsampled images. Therefore, it is still about 3× faster than COLMAP.

Table 4. Runtime (in second) of depth map generation for different methods on Strecha dataset.

dataset	#images	Gipuma	COLMAP	ACMH	ACMM
Fountain	11	235.58	1046.88	**173.55**	321.66
HerzJesu	8	134.34	709.14	**88.85**	141.26

8. Conclusion

In this work, we propose a novel multi-view stereo method for effective and efficient depth map estimation. Based on structured region information, we first present our basic MVS method with Adaptive Checkerboard sampling and Multi-Hypothesis joint view selection (ACMH). These strategies help to propagate good hypotheses as soon as possible and infer pixelwise view selection. Focusing on the depth estimation in low-textured areas, we further combine ACMH with our proposed multi-scale geometric consistency guidance scheme (ACMM). The multi-scale geometric consistency together with a detail restorer helps obtain more discrimination over low-textured areas while retaining fine details. In experiments, we demonstrate that our methods can obtain smooth and consistent depth map estimation together with complete dense 3D models while keeping a good efficiency, which shows promising applications of our methods.

Acknowledgments. This work was supported by the National Natural Science Foundation of China under Grant 61772213 and Grant 91748204.

[3] Note that, all these methods use only every other row and column in the window to compute the matching cost.

References

[1] Christian Bailer, Manuel Finckh, and Hendrik P. A. Lensch. Scale robust multi view stereo. In *Proceedings of the European Conference on Computer Vision*, pages 398–411, 2012. 1, 2, 3

[2] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman. Patchmatch: A randomized correspondence algorithm for structural image editing. In *ACM SIGGRAPH*, pages 24:1–24:11, 2009. 1

[3] D. Cremers and K. Kolev. Multiview stereo and silhouette consistency via convex functionals over convex domains. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(6):1161–1174, 2011. 2

[4] Carlos Hernndez Esteban and Francis Schmitt. Silhouette and stereo fusion for 3d object modeling. *Computer Vision and Image Understanding*, 96(3):367 – 392, 2004. 2

[5] O. Faugeras and R. Keriven. Variational principles, surface evolution, pdes, level set methods, and the stereo problem. *IEEE Transactions on Image Processing*, 7(3):336–344, 1998. 2

[6] Yasutaka Furukawa and Carlos Hernández. Multi-view stereo: A tutorial. *Found. Trends. Comput. Graph. Vis.*, 9(1-2):1–148, 2015. 2

[7] Y. Furukawa and J. Ponce. Accurate, dense, and robust multiview stereopsis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(8):1362–1376, 2010. 1, 2, 7

[8] S. Galliani, K. Lasinger, and K. Schindler. Massively parallel multiview stereopsis by surface normal diffusion. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 873–881, 2015. 1, 2, 3, 4, 5, 6, 7

[9] M. Goesele, N. Snavely, B. Curless, H. Hoppe, and S. M. Seitz. Multi-view stereo for community photo collections. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1–8, 2007. 1, 2

[10] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2 edition, 2004. 4, 5

[11] V. H. Hiep, R. Keriven, P. Labatut, and J. Pons. Towards high-resolution large-scale multi-view stereo. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1430–1437, 2009. 2

[12] X. Hu and P. Mordohai. Least commitment, viewpoint-based, multi-view stereo. In *International Conference on 3D Imaging, Modeling, Processing, Visualization Transmission*, pages 531–538, 2012. 6, 7

[13] M. Jancosek and T. Pajdla. Multi-view reconstruction preserving weakly-supported surfaces. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3121–3128, 2011. 6, 7

[14] T. Kanade and M. Okutomi. A stereo matching algorithm with an adaptive window: theory and experiment. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(9):920–932, 1994. 5

[15] Johannes Kopf, Michael F. Cohen, Dani Lischinski, and Matt Uyttendaele. Joint bilateral upsampling. *ACM Trans. Graph.*, 26(3), 2007. 5

[16] Andreas Kuhn, Heiko Hirschmüller, Daniel Scharstein, and Helmut Mayer. A tv prior for high-quality scalable multi-view stereo reconstruction. *International Journal of Computer Vision*, 124(1):2–17, 2017. 7

[17] M. Lhuillier and L. Quan. A quasi-dense approach to surface reconstruction from uncalibrated images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(3):418–433, 2005. 2

[18] Christoph Rhemann Michael Bleyer and Carsten Rother. Patchmatch stereo - stereo matching with slanted support windows. In *Proceedings of the British Machine Vision Conference*, pages 14.1–14.11, 2011. 2

[19] Johannes L. Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *Proceedings of the European Conference on Computer Vision*, pages 501–518, 2016. 1, 2, 3, 4, 5, 6, 7

[20] Thomas Schöps, Johannes L. Schönberger, Silvano Galliani, Torsten Sattler, Konrad Schindler, Marc Pollefeys, and Andreas Geiger. Eth3d benchmark. `https://www.eth3d.net`. 8

[21] T. Schöps, J. L. Schönberger, 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*, pages 2538–2547, 2017. 1, 2, 6

[22] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 519–528, 2006. 2

[23] Q. Shan, R. Adams, B. Curless, Y. Furukawa, and S. M. Seitz. The visual turing test for scene reconstruction. In *International Conference on 3D Vision*, pages 25–32, 2013. 1

[24] Q. Shan, B. Curless, Y. Furukawa, C. Hernandez, and S. M. Seitz. Occluding contours for multi-view stereo. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4002–4009, 2014. 1

[25] S. Shen. Accurate multiple view 3d reconstruction using patch-based stereo for large-scale scenes. *IEEE Transactions on Image Processing*, 22(5):1901–1914, 2013. 1

[26] S. N. Sinha, P. Mordohai, and M. Pollefeys. Multi-view stereo via graph cuts on the dual of an adaptive tetrahedral mesh. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1–8, 2007. 2

[27] C. Strecha, W. von Hansen, L. Van Gool, P. Fua, and U. Thoennessen. On benchmarking camera calibration and multi-view stereo for high resolution imagery. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008. 2, 6

[28] Radim Tylecek and R Sara. Refinement of surface mesh for accurate multiview reconstruction. *International Journal of Virtual Reality*, 9(1):45–54, 2010. 7

[29] G. Vogiatzis, C. Hernndez Esteban, P. H. S. Torr, and R. Cipolla. Multiview stereo via volumetric graph-cuts and oc-

clusion robust photo-consistency. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(12):2241–2246, 2007. 2

[30] Jian Wei, Benjamin Resch, and Hendrik Lensch. Multi-view depth map estimation with cross-view consistency. In *Proceedings of the British Machine Vision Conference*, 2014. 2, 3

[31] Kuk-Jin Yoon and In-So Kweon. Locally adaptive support-weight approach for visual correspondence search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 924–931 vol. 2, 2005. 5

[32] Christopher Zach. Fast and high quality fusion of depth maps. In *International Conference on 3D Imaging, Modeling, Processing, Visualization Transmission*, 2008. 7

[33] A. Zaharescu, E. Boyer, and R. Horaud. Topology-adaptive mesh deformation for surface evolution, morphing, and multiview reconstruction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(4):823–837, 2011. 7

[34] Guofeng Zhang, Jiaya Jia, Tien-Tsin Wong, and Hujun Bao. Recovering consistent video depth maps via bundle optimization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008. 5

[35] K. Zhang, J. Lu, and G. Lafruit. Cross-based local stereo matching using orthogonal integral images. *IEEE Transactions on Circuits and Systems for Video Technology*, 19(7):1073–1079, 2009. 5

[36] E. Zheng, E. Dunn, V. Jojic, and J. M. Frahm. Patchmatch based joint view selection and depthmap estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1510–1517, 2014. 1, 2, 3, 7