Machine Vision and Applications (2023) 34:32
https://doi.org/10.1007/s00138-023-01380-8

ORIGINAL PAPER

PM-MVS: PatchMatch multi-view stereo

Koichi Ito[1] · Takafumi Ito[1] · Takafumi Aoki[1]

Received: 11 April 2021 / Revised: 27 December 2022 / Accepted: 26 January 2023 / Published online: 2 March 2023

Abstract
PatchMatch Stereo is a method for generating a depth map from stereo images by repeating spatial propagation and view propagation. The concept of PatchMatch Stereo can be easily extended to Multi-View Stereo (MVS). In this paper, we present PatchMatch Multi-View Stereo (PM-MVS), which is a highly accurate 3D reconstruction method that can be used in various environments. Three techniques are introduced to PM-MVS: (i) matching score evaluation, (ii) viewpoint selection, and (iii) outlier filtering. The combination of normalized cross-correlation with bilateral weights and geometric consistency between viewpoints is used to improve the estimation accuracy of depth and normal maps at object boundaries and poor-texture regions. For each pixel, viewpoints used for stereo matching are carefully selected in order to improve robustness against disturbances such as occlusion, noise, blur, and distortion. Outliers are removed from reconstructed 3D point clouds by a weighted median filter and consistency-based filters assuming multi-view geometry. Through a set of experiments using public multi-view image datasets, we demonstrate that the proposed method exhibits efficient performance compared with conventional methods.

Keywords PatchMatch · 3D reconstruction · Multi-view stereo

1 Introduction

Stereo matching is a technique to find correspondence between two images captured by a stereo camera, and is one of fundamental processes in image processing and computer vision [17,22]. The 3D shape of a target object can be reconstructed from the correspondence obtained by stereo matching, considering the geometric relationship between cameras. Multi-View Stereo (MVS), which uses a set of images taken from multiple viewpoints for dense reconstruction of the target object, has been widely studied [22].

PatchMatch Stereo proposed by Bleyer et al. [1] is one of convincing stereo matching methods. PatchMatch Stereo generates a disparity map (and a normal map) from a binocular stereo image pair by repeatedly updating the disparity and normal maps, which are initialized with random values in advance. The update process consists of three steps: (i) *spatial propagation*, (ii) *view propagation*, and (iii) *plane refinement*. PatchMatch Stereo exhibits efficient performance with fewer stereo matches than brute-force matching approach by introducing realistic assumptions that take into account the characteristics of disparity map. PatchMatch Stereo can also estimate the disparity between stereo images with sub-pixel accuracy. In addition, PatchMatch Stereo estimates the normal of each pixel, enabling robust 3D reconstruction against local image deformation. With these advantages, PatchMatch Stereo is expected to become one of the most effective stereo matching methods for 3D reconstruction.

The concept of PatchMatch Stereo can be easily extended to MVS. Shen proposed a multi-view 3D reconstruction method based on PatchMatch Stereo [21]. Shen's method is very ad hoc and does not take full advantage of the potential of multi-view images; the method simply combines a set of depth maps, each derived from a pair of stereo images. On the other hand, it is well known in the field of MVS that the robustness and accuracy of 3D reconstruction from multi-view images can be improved by integrating matching scores from multiple stereo image pairs [4,6,16,22]. This approach *matching score integration* could be applied to derive an efficient multi-view extension of Bleyer's original PatchMatch Stereo [1].

In line with this idea, Schönberger et al. proposed COLMAP [19], which uses a matching score that takes

✉ Koichi Ito
ito@aoki.ecei.tohoku.ac.jp

[1] Graduate School of Information Sciences, Tohoku University, 6-6-05, Aramaki Aza Aoba, Sendai-shi 980-8579, Japan

into account multi-view integration unlike Shen's method. COLMAP estimates depth and normal maps by introducing a hidden Markov model to the parameter update algorithm for PatchMatch Stereo. COLMAP is one of the most accurate multiview 3D reconstruction algorithms. On the other hand, a major concern of COLMAP is that it simplifies the depth/normal update process to reduce computational complexity. *Spatial propagation*, which is to propagate depth and normal parameters to neighboring pixels, is performed only on the pixels one pixel adjacent to the pixel of interest. In addition, *view propagation*, which is used in the original PatchMatch Stereo to propagate parameters to another viewpoint, is not used in COLMAP. *Plane refinement*, which is the updating of parameters using random numbers, is performed multiple times in the original PatchMatch Stereo, but only once in COLMAP. Such ad hoc simplification could degrade overall 3D reconstruction performance.

In our work [9] published earlier than COLMAP, we proposed a systematic extension of PatchMatch Stereo taking the multi-view integration into consideration. This method is different from Shen's method in the following points: (i) depth maps are updated with interaction among multi-view images, (ii) matching score is calculated from multiple stereo image pairs, and (iii) *view propagation* is also performed among multi-view images. In this method, however, the viewpoints used for matching is selected roughly for each reference viewpoint and not for each pixel, so the reconstruction accuracy may be degraded by image occlusion and noise. The estimation accuracy of depth and normal maps at object boundaries and poor-texture regions may also be degraded since simple Normalized Cross-Correlation (NCC) is used as a measure of matching. The estimated depth and normal maps are used directly to reconstruct the object shape, so the result will be significantly affected by the areas where the estimation failed, resulting in outliers and missing points.

In this paper, we propose PatchMatch Multi-View Stereo (PM-MVS), a highly accurate 3D reconstruction method addressing the above problems and can be used in various environments. We introduce three improvement techniques into PM-MVS, related to (i) matching score evaluation, (ii) viewpoint selection, and (iii) outlier filtering. For (i), we employ NCC with bilateral weights as an advanced matching measure and reflect geometric consistency for each stereo pair to improve robustness of matching. For (ii), we modify the algorithm so that the viewpoint used to calculate the matching score can be selected for each pixel. For (iii), we remove outliers by a weighted median filter and three specially designed filters based on the consistency of multi-view geometry [26]. Through a set of experiments using public multi-view image datasets, we demonstrate that the proposed method exhibits efficient performance compared with conventional methods.

2 Related work

In the following, we briefly summarize well-known multi-view 3D reconstruction algorithms that are also used for performance comparison with the proposed method.

The MVS algorithms based on region expansion reconstruct the 3D shape by performing 3D reconstruction of feature points and then repeatedly propagating the results to neighboring regions [4,8,12]. One of the most well-known methods is Patch-based Multi-View Stereo (PMVS) [4]. PMVS reconstructs a sparse 3D shape based on feature points detected in the input image, and then reconstructs a dense 3D shape by repeating propagation of the reconstruction result and filtering based on consistency of visibility. Algorithms based on region expansion have the advantages of fast processing and not requiring the 3D reconstruction results obtained by other methods as initial values. There are some problems that the entire object cannot be reconstructed due to a small number of feature points, and that the reconstruction accuracy is degraded in regions where no feature points are detected since these algorithms propagate the sparse results reconstructed from the feature points. It is also difficult to reconstruct areas with small changes in intensity, such as poor-texture areas. In many cases, outliers are included in the reconstruction results from feature points, and it is important to remove them in order to perform stable 3D reconstruction.

The MVS algorithms based on depth map integration estimate depth maps for each viewpoint from multi-view images, and then integrate them to reconstruct the 3D shape of the target [2,6,13,19,23]. Depth is estimated by calculating the likelihood of the assumed depth using image matching such as NCC, and then a 3D point cloud or 3D mesh model is generated by integrating the depth maps generated for each viewpoint with consistency. Goesele et al. [6] used NCC-based window matching in the framework of plane-sweeping approach to generate highly accurate depth maps. Campbell et al. [2] assigned multiple depth candidates to a single pixel and selected the best depth based on the information of neighboring pixels, resulting in a highly accurate 3D shape reconstruction. Tola et al. [23] used DAISY descriptors [3] to improve the robustness against stereo images with large image deformations. Schönberger et al.[19] proposed COLMAP for fast and accurate 3D reconstruction by combining a hidden Markov model with the parameter update algorithm used in PatchMatch Stereo [1]. Goesele et al.'s method and Campbell et al.'s method are based on a plane-sweeping approach, which requires a full search in the depth direction to estimate the depth corresponding to a pixel. Therefore, these methods are not practical in terms of computational cost because of the large number of window matching calculations. Tola et al.'s method and COLMAP can reconstruct the shape with short processing time and high accuracy, however, a sparse 3D shape is reconstructed depending on the

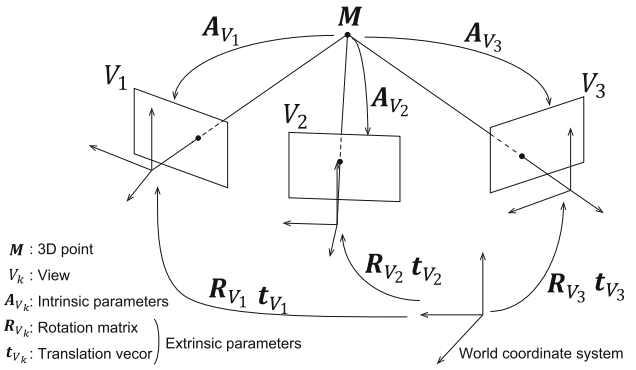

Fig. 1 Geometric relationship among the 3D point $\boldsymbol{M}$ and views V_k $(k = 3)$

object since they achieve high accuracy of 3D reconstruction by excluding points with low confidence values.

In our previous work [9], we proposed an extension of PatchMatch stereo [1] to MVS as well as COLMAP. In this method, depth maps are updated with interaction among multi-view images, a matching score is calculated from multiple stereo images, and view propagation is performed among multi-view images. The reconstruction accuracy of this method can be improved by filtering based on the consistency of multi-view geometry [26]. As mentioned in Sect. 1, the reconstruction accuracy is highly dependent on the environment since the viewpoints used for matching are selected for each viewpoint and NCC is used for matching among multi-view images.

3 Fundamental techniques for PM-MVS

This section describes fundamental techniques for PM-MVS: (i) matching score, (ii) viewpoint selection, and (iii) outlier filtering. We use the following notations to describe each technique. We now consider a set of views $\boldsymbol{V} = \{V_1, V_2, \ldots, V_K\}$. For each view $V_k \in \boldsymbol{V}$, let $I_{V_k}(\boldsymbol{m})$ be a reference image, $\boldsymbol{A}_{V_k}$ be the intrinsic parameters, and $\boldsymbol{R}_{V_k}$ and $\boldsymbol{t}_{V_k}$ be the extrinsic parameters consisting of a rotation matrix and a translation vector. K is the number of images and $\boldsymbol{m} = (u, v)$ is an image coordinate. We consider the problem of generating depth maps $d_{V_k}(\boldsymbol{m})$ and normal maps $\theta_{V_k}(\boldsymbol{m})$ and $\phi_{V_k}(\boldsymbol{m})$ for all the views in $\boldsymbol{V}$. $\theta_{V_k}(\boldsymbol{m})$ and $\phi_{V_k}(\boldsymbol{m})$ indicate the angle of X-axis direction and Y-axis direction of the normal vector, respectively. Note that we use d_{V_k}, θ_{V_k}, and ϕ_{V_k} for $d_{V_k}(\boldsymbol{m})$, $\theta_{V_k}(\boldsymbol{m})$, and $\phi_{V_k}(\boldsymbol{m})$, respectively, unless necessary in the following. Figure 1 shows geometric relationship among views and a target object when $k = 3$.

3.1 Matching score

We employ a confidence value proposed by Goesele et al. [6] as a matching score to utilize multiple stereo images in the framework of PM-MVS. In the most MVS algorithms [4,6,21], NCC is used to evaluate the matching of multi-view images. NCC-based matching produces wrong correspondence at object boundaries and in poor-texture regions, resulting in the estimation of discontinuous depths and normals, which cause outliers. Filtering of the 3D point cloud removes some outliers, however it cannot remove them completely, which reduces the reconstruction accuracy. The matching score in PM-MVS is based on BNCC, which is NCC with bilateral weights, used in COLMAP [19]. The differences between PM-MVS and COLMAP are as follows. The matching score in PM-MVS is obtained by subtracting a penalty calculated based on the geometric consistency between viewpoints from the similarity between windows calculated by BNCC. Also, the average of the matching scores of the top-L stereo pairs out of all stereo pairs is used to suppress the effect of occlusion. In the following, we revise the definition of BNCC and provide details on the mathematical definitions of the matching scores used in PM-MVS.

We consider the matching score for the reference view $V_k \in \boldsymbol{V}$ in the following. Let us assume that $\boldsymbol{C}_{V_k} = \{C^n_{V_k} | n = 1, \ldots, N_{pair}\}$ is a set of stereo pairs to be matched with V_k, where N_{pair} is the number of stereo pairs. As described in Sect. 3.2, each $\boldsymbol{m}$ has a different viewpoint to be paired, and therefore, $C^n_{V_k}$ should be written as $C^n_{V_k}(\boldsymbol{m})$ to be precise. In the following, we use the notation $C^n_{V_k}$ for ease of understanding. Given a pixel $\boldsymbol{m}$ in V_k and parameter $\boldsymbol{p}_{V_k} = \{d_{V_k}, \theta_{V_k}, \phi_{V_k}\}$, a matching score $\xi(V_k, C^n_{V_k}, \boldsymbol{p}_{V_k}, \boldsymbol{m})$ between V_k and $C^n_{V_k}$ is defined by

$$\xi(V_k, C^n_{V_k}, \boldsymbol{p}_{V_k}, \boldsymbol{m}) = \text{BNCC}(f, g) - \psi(V_k, C^n_{V_k}, \boldsymbol{p}_{V_k}, \boldsymbol{m}), \tag{1}$$

where BNCC is NCC with bilateral weights, which is defined by

$$\text{BNCC}(f, g) = \frac{\sum_i b_i (f_i - \bar{f}^*)(g_i - \bar{g}^*)}{\sqrt{\sum_i b_i (f_i - \bar{f}^*) \sum_i b_i (g_i - \bar{g}^*)}}. \tag{2}$$

f and g are defined by

$$f = \text{Crop}(I_{V_k}, \boldsymbol{m}, w), \tag{3}$$

$$g = \text{Crop}(\text{Trans}(I_{C^n_{V_k}}, \boldsymbol{H}(V_k, C^n_{V_k}, \boldsymbol{p}_{V_k}, \boldsymbol{m})), \boldsymbol{m}, w), \tag{4}$$

where $\text{Crop}(I, \boldsymbol{m}, w)$ indicates a function to crop a window with $w \times w$ pixels centered on the coordinate $\boldsymbol{m}$ from the image I. $\text{Trans}(I, \boldsymbol{H})$ indicates a function to transform I using a projective matrix $\boldsymbol{H}$.Given parameters $\boldsymbol{p}_{V_k} =$

$\{d_{V_k}, \theta_{V_k}, \phi_{V_k}\}$, the projective matrix $\boldsymbol{H}$ between $V_k - C^n_{V_k}$ is defined by

$$\boldsymbol{H}(V_k, C^n_{V_k}, \boldsymbol{p}_{V_k}, \boldsymbol{m}) = \boldsymbol{A}_{C^n_{V_k}} \left(\boldsymbol{R} + \frac{\boldsymbol{t}\boldsymbol{n}^T}{\boldsymbol{n}^T \boldsymbol{M}} \right) \boldsymbol{A}^{-1}_{V_k}, \tag{5}$$

where a rotation matrix $\boldsymbol{R}$, a translation vector $\boldsymbol{t}$, a 3D coordinate $\boldsymbol{M}$ and a normal vector $\boldsymbol{n}$ are defined by

$$\begin{aligned}
\boldsymbol{R} &= \boldsymbol{R}_{C^n_{V_k}} \boldsymbol{R}^{-1}_{V_k}, \\
\boldsymbol{t} &= \boldsymbol{t}_{C^n_{V_k}} - \boldsymbol{R}_{C^n_{V_k}} \boldsymbol{R}^{-1}_{V_k} \boldsymbol{t}_{V_k}, \\
\boldsymbol{M} &= d_{V_k} \boldsymbol{A}^{-1}_{V_k} [u, v, 1]^T, \\
\boldsymbol{n} &= \frac{1}{\sqrt{\tan^2 \theta_{V_k} + \tan^2 \phi_{V_k} + 1}} [\tan \theta_{V_k}, \tan \phi_{V_k}, -1]^T,
\end{aligned}$$

respectively. In Eq. (2), i indicates a pixel in the windows. $\bar{f}^*$ and $\bar{g}^*$ indicate the weighted average calculated using the pixel values and weights b_i of each window. The bilateral weight b_i at pixel i is defined by

$$b_i = \exp\left(-\frac{|f_i - f_c|^2}{2\sigma_f^2} - \frac{\|\boldsymbol{m}_i - \boldsymbol{m}_c\|_2^2}{2\sigma_m^2} \right), \tag{6}$$

where the subscript c indicates the center coordinate of the window. $|f_i - f_c|^2$ indicates the pixel value distance and $\|\boldsymbol{m}_i - \boldsymbol{m}_c\|_2^2$ indicates the spatial distance, whose importance is relatively scaled by their Gaussian dispersion σ_f and σ_m.

$\psi(V_k, C^n_{V_k}, \boldsymbol{p}_{V_k}, \boldsymbol{m})$ in Eq. (1) indicates the geometric consistency between V_k and $C^n_{V_k}$ at pixel $\boldsymbol{m}$ on V_k. In poor-texture or noisy regions, the scores obtained by BNCC are less reliable. Therefore, adding geometric consistency as a penalty improves the reliability of the matching scores for such regions. The geometric consistency is defined by the reprojection error $\Delta e(\boldsymbol{m})$ between V_k and $C^n_{V_k}$ as in [19]. The 3D point $\boldsymbol{M}$ for $\boldsymbol{m}$ on V_k is calculated by

$$\boldsymbol{M} = \boldsymbol{R}^{-1}_{V_k} (d_{V_k}(\boldsymbol{m}) \cdot \boldsymbol{A}^{-1}_{V_k} [u, v, 1]^T) - \boldsymbol{R}^{-1}_{V_k} \boldsymbol{t}_{V_k}. \tag{7}$$

$\boldsymbol{M}$ is projected onto $C^n_{V_k}$ by

$$\boldsymbol{m}' = \boldsymbol{A}_{C^n_{V_k}} [\boldsymbol{R}_{C^n_{V_k}} \boldsymbol{t}_{C^n_{V_k}}] \boldsymbol{M} \tag{8}$$

as shown in Fig. 2 (a). Then, the 3D point $\boldsymbol{M}'$ for $\boldsymbol{m}' = (u', v')$ on $C^n_{V_k}$ is calculated by

$$\boldsymbol{M}' = \boldsymbol{R}^{-1}_{C^n_{V_k}} (d_{C^n_{V_k}}(\boldsymbol{m}') \cdot \boldsymbol{A}^{-1}_{C^n_{V_k}} [u', v', 1]^T) - \boldsymbol{R}^{-1}_{C^n_{V_k}} \boldsymbol{t}_{C^n_{V_k}}. \tag{9}$$

$\boldsymbol{M}'$ is projected onto V_k by

$$[\hat{u}, \hat{v}, 1]^T = \boldsymbol{A}_{V_k} [\boldsymbol{R}_{V_k} \boldsymbol{t}_{V_k}] \boldsymbol{M}' \tag{10}$$

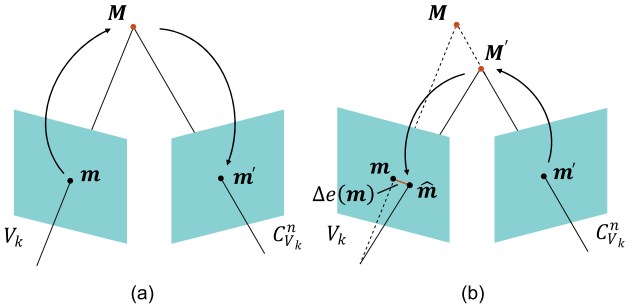

Fig. 2 Illustration of reprojection error $\Delta e(\boldsymbol{m})$: **a** $\boldsymbol{m}'$ is obtained by projecting a 3D point M onto $C^n_{V_k}$, which is reconstructed using $\boldsymbol{m}$ on V_k and its parameters and **b** $\hat{\boldsymbol{m}}$ is obtained by projecting a 3D point M' on to V_k, which is reconstructed using $\boldsymbol{m}'$ on $C^n_{V_k}$ and its parameters. The reprojection error $\Delta e(\boldsymbol{m})$ is calculated as the distance between $\boldsymbol{m}$ and $\hat{\boldsymbol{m}}$

as shown in Fig. 2b. The reprojection error is given by

$$\Delta e(\boldsymbol{m}) = \|\boldsymbol{m} - \hat{\boldsymbol{m}}\|_2, \tag{11}$$

where $\hat{\boldsymbol{m}} = (\hat{u}, \hat{v})$. The geometric consistency is given by

$$\psi(V_k, C^n_{V_k}, \boldsymbol{p}_{V_k}, \boldsymbol{m}) = \eta \min(\Delta e(\boldsymbol{m}), \psi_{max}), \tag{12}$$

where ψ_{max} indicates the maximum of the acceptable reprojection error and η indicates the constant.

We obtain a set of matching scores by calculating the matching score for all the stereo pairs. The effect of occlusion can be reduced by considering the top-L matching scores [5]. Assuming that the matching score sorted in descending order is $\hat{\xi}(V_k, C^n_{V_k}, \boldsymbol{p}_{V_k}, \boldsymbol{m})$, the final matching score for pixel $\boldsymbol{m}$ on the reference view V_k is calculated by

$$Score(V_k, \boldsymbol{C}_{V_k}, \boldsymbol{p}_{V_k}, \boldsymbol{m}) = \frac{1}{L} \sum_{l=1}^{L} \hat{\xi}(V_k, C^l_{V_k}, \boldsymbol{p}_{V_k}, \boldsymbol{m}). \tag{13}$$

3.2 Viewpoint selection

The original approaches of MVS with PatchMatch [9,21] select one of the viewpoints, $C^n_{V_k}$, to make a stereo pair and match all the pixels in the image of $C^n_{V_k}$ with those of the reference viewpoint V_k. Since it is assumed that the pixels of V_k correspond to those of $C^n_{V_k}$, the accuracy of depth and normal estimation is degraded by disturbances such as occlusion and noise. Therefore, the optimal viewpoint $C^n_{V_k}$ to be matched with V_k has to be selected for each pixel, not for each viewpoint, as used in recent approaches [15,19,24,27] to improve the matching accuracy. In the proposed method, three metrics are introduced for pixel-wise viewpoint selection: (i) matching score, (ii) triangulation probability, and (iii) incident probability. Our approach is similar to Goesele et al.

[7], although it is not pixel-wise viewpoint selection. Both approaches use convergence angles between viewpoints and an NCC-based score. In [7], the number of SIFT features shared among viewpoints and image resolution are used. On the other hand, the proposed approach uses normals and a mesh generated from a sparse 3D points obtained by SfM.

3.2.1 Matching score

Generally, the viewpoints are selected in the order of shortest baseline length to make a stereo pair with less image deformation, however, the effects of occlusion and noise are not taken into account in this case. To robustly select viewpoints against noise and occlusion, we employ a metric based on the matching score, which is defined by

$$P_{score} = \exp\left(-\frac{\{1-\xi(V_k, C^j_{V_k}, \boldsymbol{p}_{V_k}, \boldsymbol{m})\}^2}{2\sigma_s^2}\right), \tag{14}$$

where $\xi(V_k, C^j_{V_k}, \boldsymbol{p}_{V_k}, \boldsymbol{m})$ indicates a matching score defined in Eq. (1). $C^j_{V_k}$ indicates the j-th viewpoint among the N_s viewpoints. σ_s is a parameter of the Gaussian function and is the threshold for determining whether a window extracted from V_k is included in $C^j_{V_k}$.

3.2.2 Triangulation probability

The matching score is high if the intensity values between the windows are correlated. In general, windows extracted from viewpoints with a short baseline length with V_k exhibit a high correlation since the image deformation between viewpoints is small. Note that when the angle between viewpoints is close to zero, the windows are highly correlated with each other for any given depth, resulting in inaccurate depth estimation. In order to avoid this problem and improve the accuracy of viewpoint selection, the triangulation prior P_{tri} proposed in [19] is introduced to the proposed method, which is defined by

$$P_{tri} = 1 - \frac{\{\min(\theta_{tri}, \tau_{tri}) - \tau_{tri}\}^2}{\tau_{tri}^2}, \tag{15}$$

where τ_{tri} is a threshold and θ_{tri} indicates a triangulation angle between viewpoints as shown in Fig. 3, which is given by

$$\theta_{tri} = \arccos\frac{(\boldsymbol{M} - \boldsymbol{O}_{C^j_{V_k}})^T \cdot \boldsymbol{M}}{\|\boldsymbol{M} - \boldsymbol{O}_{C^j_{V_k}}\|_2 \|\boldsymbol{M}\|_2}, \tag{16}$$

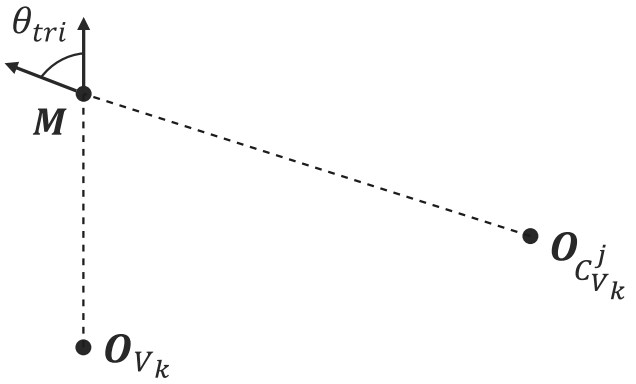

Fig. 3 Illustration of the triangulation angle θ_{tri}, where $\boldsymbol{M}$ is a 3D point reconstructed from the depth $d_{V_k}(\boldsymbol{m})$, $\boldsymbol{O}_{V_k}$ indicates the camera center of V_k, and $\boldsymbol{O}_{C^j_{V_k}}$ indicates the camera center of $C^j_{V_k}$

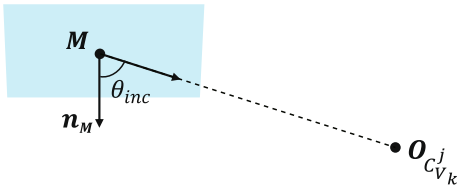

Fig. 4 Illustration of the incident angle θ_{inc}, where $\boldsymbol{M}$ is a 3D point reconstructed from the depth $d_{V_k}(\boldsymbol{m})$, $\boldsymbol{n}_M$ indicates the normal vector $\boldsymbol{n}_M$ of the 3D point $\boldsymbol{M}$, and $\boldsymbol{O}_{C^j_{V_k}}$ indicates the camera center of $C^j_{V_k}$

where $\boldsymbol{M}$ is a 3D point reconstructed from the depth $d_{V_k}(\boldsymbol{m})$, and $\boldsymbol{O}_{C^j_{V_k}}$ indicates the camera center of V^j_k. P_{tri} is low when the triangulation angle θ_{tri} is below the threshold τ_{tri}.

3.2.3 Incident probability

If the normal vector $\boldsymbol{n}_M$ of the 3D point $\boldsymbol{M}$ and the eye vector of the viewpoint $C^j_{V_k}$ have the same direction, $\boldsymbol{M}$ is not visible in $C^j_{V_k}$. In order to exclude such viewpoints and improve the accuracy of viewpoint selection, the incident prior P_{inc} proposed in [19] is introduced to the proposed method, which is defined by

$$P_{inc} = \exp\left(-\frac{\theta_{inc}^2}{2\sigma_i^2}\right), \tag{17}$$

where σ_i is a parameter of the Gaussian function, and θ_{inc} indicates an incident angle between $\boldsymbol{n}_M$ and the eye vector of the viewpoint $C^j_{V_k}$ as shown in Fig. 4, which is given by

$$\theta_{inc} = \arccos\frac{(\boldsymbol{O}_{C^j_{V_k}} - \boldsymbol{M})^T \cdot \boldsymbol{n}_M}{\|\boldsymbol{O}_{C^j_{V_k}} - \boldsymbol{M}\|_2 \|\boldsymbol{n}_M\|_2}. \tag{18}$$

The above three metrics are used to select a set of viewpoints $\boldsymbol{C}_{V_k}$ to be paired with the reference viewpoint V_k for each pixel $\boldsymbol{m}$. The score $P(V_k, C^j_{V_k}, \boldsymbol{p}_{V_k}, \boldsymbol{m})$ for each viewpoint $C^j_{V_k}$ is calculated by

$$P(V_k, C^j_{V_k}, \boldsymbol{p}_{V_k}, \boldsymbol{m}) = P_{score} \cdot P_{tri} \cdot P_{inc}, \tag{19}$$

where $C^j_{V_k}$ indicates the j-th viewpoint in $\boldsymbol{C}_{V_k}$. A set of viewpoints $\boldsymbol{C}_{V_k}$ consists of N_s viewpoints in order of decreasing baseline length to V_k. We limit the number of viewpoints to N_s instead of all viewpoints in viewpoint selection to eliminate distant viewpoints and reduce the number of candidate viewpoints for reducing the processing time. Since PM-MVS is an iterative method, the accuracy of depth and normal is low at first. The accuracy of the viewpoint selection score calculated in Eq. (19) is also low, resulting in inaccurate estimation of depth and normal maps. Therefore, the sparse 3D point cloud obtained by Structure from Motion (SfM) used in the estimation of camera parameters is used. A mesh model is generated from the sparse 3D point cloud using Poisson surface reconstruction [11], and the depth and normal maps corresponding to the reference viewpoint V_k are rendered from its mesh model. Equation (19) is calculated for the depth and normal from $\boldsymbol{p}_{V_k}$ and from the sparse 3D point cloud, respectively, and the larger value is used as the score for viewpoint selection. The viewpoint $C^j_{V_k}$ corresponding to the top N_{pair} of $P(V_k, C^j_{V_k}, \boldsymbol{p}_{V_k}, \boldsymbol{m})$ is selected as a set of viewpoints $\boldsymbol{C}_{V_k}(\boldsymbol{m})$ that should be paired to estimate the parameters of pixel $\boldsymbol{m}$ in V_k.

3.3 Filtering

The depth map and normal map estimated by MVS have wrong correspondence in poor texture regions and object boundaries, and these result in outliers and missing points in 3D reconstruction. It is necessary to remove or interpolate such wrong correspondence in depth and normal maps to obtain highly accurate reconstruction results. The proposed method uses a weighted median filter and three filters based on the consistency of multi-view geometry [26] to suppress the occurrence of outliers and missing points in the reconstruction results.

3.3.1 Weighted median filter

A weighted median filter [22] has been used to improve the accuracy of disparity estimation in stereo vision [14] and depth and normal estimation in MVS. The weighted median filter is introduced into PM-MVS not only to remove outliers, but also to interpolate missing points. In the proposed method, the weight for the weighted median filter is calculated from the matching score and bilateral weights. The weight $w_{med}(\boldsymbol{m})$ on $\boldsymbol{m}$ is calculated by

$$w_{med}(\boldsymbol{m}) = b_i \exp\left(-\frac{1 - Score(V_k, \boldsymbol{p}_{V_k}, \boldsymbol{m})^2}{2\sigma_x^2}\right). \tag{20}$$

3.3.2 Consistency among depth maps and their visibility

This filter checks consistency among the multiple depth maps and their visibility. If a 3D point interrupts the visibility of other 3D points or its visibility is interrupted by other 3D points, this point is removed as an outlier.

3.3.3 Left-right consistency

This filter is similar to left-right consistency checking used in binocular stereo matching. We remove a point whose distance from each corresponding point in all other views is longer than threshold, where we use the depth instead of the distance.

3.3.4 Consistency of pixel intensity

This filter checks the consistency of pixel intensity among the multiple images to remove artifacts observed around the surface. We do not take care of a 3D point near other points in the filter described in Sect. 3.3.2, since it is hard to check the consistency of such a 3D point using only geometric relation. The use of pixel intensity makes it possible to classify such a 3D point into a true 3D point or an outlier.

For more details on the above four filters, refer to Yodokaw et al. [26].

4 PatchMatch multi-view stereo (PM-MVS)

The proposed method consists of four steps: (i) initialization, (ii) spatial propagation, (iii) view propagation, and (iv) plane refinement. The flow of PM-MVS for reference view V_k is shown in Fig. 5. Depth and normal maps are generated by repeating processes (ii)–(iv). The processing flow of PM-MVS follows that of [1], except that the viewpoint is updated at each iteration, although the content of each process is different. The detail of each step in PM-MVS is described in the following.

4.1 Initialization

This step consists of parameter initialization by random numbers, viewpoint selection, and calculation of the initial matching score.

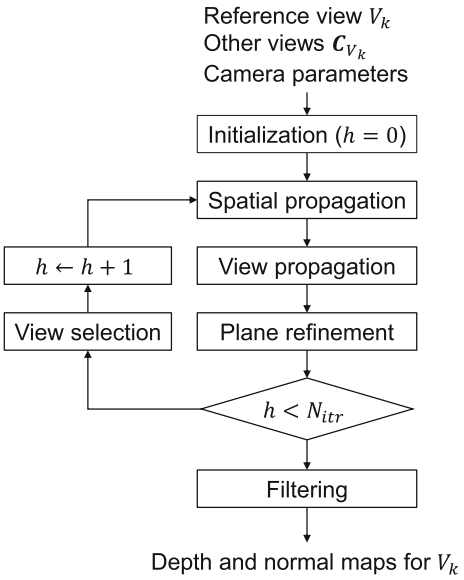

Fig. 5 Flow of the proposed method for the reference view V_k

4.1.1 Parameter initialization by random numbers

In 3D reconstruction methods using PatchMatch, the values of depth and normal maps are initialized by random numbers. It is necessary to set the appropriate range of random numbers since the range of the random numbers corresponds to the reconstruction range. In the proposed method, we employ the difference approaches for setting the range of random numbers depending on whether SfM is used to estimate the camera parameters or not.

In the case of using SfM, the camera parameters, i.e., the intrinsic and extrinsic parameters of the cameras, are estimated and the sparse 3D point cloud is reconstructed simultaneously. The 3D point could is projected onto the reference viewpoint V_k to obtain a set of depth $\boldsymbol{Z}_{V_k}$. Since $\boldsymbol{Z}_{V_k}$ includes the depth from outliers, the range of depth $\Delta \boldsymbol{d}_{V_k}$ is determined by

$$\Delta \boldsymbol{d}_{V_k} = [Z_{\min}, Z_{\max}], \tag{21}$$

where $Z_{\min}$ and $Z_{\max}$ are calculated by

$$Z_{\min} = \lambda_{\min} \min{}'(\boldsymbol{Z}_{V_k}, \lfloor c_{\min} N_{\boldsymbol{Z}_{V_k}} + 1 \rfloor), \tag{22}$$

$$Z_{\max} = \lambda_{\max} \min{}'(\boldsymbol{Z}_{V_k}, \lfloor c_{\max} N_{\boldsymbol{Z}_{V_k}} + 1 \rfloor), \tag{23}$$

where $\lfloor x \rfloor$ indicates the function to round the element of x to the nearest integer towards minus infinity, $N_{\boldsymbol{Z}_{V_k}}$ is the number of elements in $\boldsymbol{Z}_{V_k}$, $\min{}'(\boldsymbol{x}, i)$ indicates the function to get the i-th smallest element in $\boldsymbol{x}$, and $\lambda_{\min}, \lambda_{\max}, c_{\min}$, and $c_{\max}$ are parameters. We employ $\{\lambda_{\min}, \lambda_{\max}, c_{\min}, c_{\max}\} = \{0.75, 1.25, 0.01, 0.99\}$ in this paper.

In the case where the camera parameters for each viewpoint are given in advance and SfM is not used, the depth range is determined using the geometric relationship between the reference viewpoint V_k and other viewpoints $\boldsymbol{C}_{V_k}$ as shown in Fig. 6. Let L be the line of sight through the image center of V_k. For each view $C^n_{V_k}$ in $\boldsymbol{C}_{V_k}$, L_{C_n} is obtained by projecting L onto the viewpoint $C^n_{V_k}$. The coordinate $\boldsymbol{x}_1^{C^n_{V_k}}$ and $\boldsymbol{x}_2^{C^n_{V_k}}$ are defined as the coordinate locating at 1/4 of the image size in this paper. Assuming that the image center of V_k corresponds to $\boldsymbol{x}_1^{C^n_{V_k}}$ and $\boldsymbol{x}_2^{C^n_{V_k}}$ on $C^n_{V_k}$, the depth $Z_{\min}^{C^n_{V_k}}$ and $Z_{\max}^{C^n_{V_k}}$ are calculated. The range of depth is set to

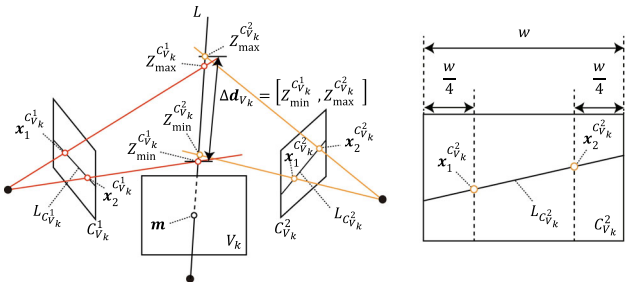

Fig. 6 Depth map initialization using the geometric relationship between the reference viewpoint V_k and other viewpoints $\boldsymbol{C}_{V_k}$

$$\Delta \boldsymbol{d}_{V_k} = [Z_{\min}, Z_{\max}], \tag{24}$$

where

$$Z_{\min} = \min\{Z_{\min}^{C^n_{V_k}} | C^n_{V_k} \in \boldsymbol{C}_{V_k}\}, \tag{25}$$

$$Z_{\max} = \max\{Z_{\max}^{C^n_{V_k}} | C^n_{V_k} \in \boldsymbol{C}_{V_k}\}. \tag{26}$$

In both cases, the range of normal is set to $\pm\pi/3$. Thus, we obtain the initial parameters $\boldsymbol{p}_{V_k} = \{d_{V_k}, \theta_{V_k}, \phi_{V_k}\}$.

4.1.2 Viewpoint selection

According to the procedure of viewpoint selection mentioned in Sect. 3.2, we obtain a set of viewpoints $C_{V_k} = \{C^n_{V_k} | n = 1, \ldots, N_{pair}\}$ from which to calculate the matching score for each pixel in the reference viewpoint V_k.

4.1.3 Calculation of initial matching scores

The above processes determine parameters and viewpoints to be used for each pixel in V_k, and the initial matching scores are calculated according to Sect. 3.1.

4.2 Spatial propagation

This step propagates the depth and normal information in the reference viewpoint V_k. As mentioned above, let $\boldsymbol{p}_{V_k}(\boldsymbol{m}) = \{d_{V_k}(\boldsymbol{m}), \theta_{V_k}(\boldsymbol{m}), \phi_{V_k}(\boldsymbol{m})\}$ be parameters for the

pixel coordinate $\boldsymbol{m} = (u, v)$ in V_k. Parameters $\boldsymbol{p}_{V_k}(\boldsymbol{m})$ are updated by comparing a matching score on the image coordinate $\boldsymbol{m}$ with matching scores on its neighboring pixels. If $Score(V_k, \boldsymbol{C}_{V_k}, \boldsymbol{p}_{V_k}(u+\delta, v), \boldsymbol{m}) > Score(V_k, \boldsymbol{C}_{V_k}, \boldsymbol{p}_{V_k}(u, v), \boldsymbol{m})$, then the parameters for (u, v) are replaced by the parameters for $(u+\delta, v)$. Similarly, if $Score(V_k, \boldsymbol{C}_{V_k}, \boldsymbol{p}_{V_k}(u, v+\delta), \boldsymbol{m}) > Score(V_k, \boldsymbol{C}_{V_k}, \boldsymbol{p}_{V_k}(u, v), \boldsymbol{m})$, then the parameters for (u, v) are replaced by the parameters for $(u, v+\delta)$. If the iteration count is odd, then spatial propagation is performed from the top-left pixel to the bottom-right pixel. Otherwise, spatial propagation is performed in the reverse order. Thus, δ indicates 1 when the iteration count is odd and -1 when the iteration count is even. The above process is performed for all the pixels in V_k.

4.3 View propagation

This step propagates the depth and normal information from the reference viewpoint V_k to the neighboring viewpoints $\boldsymbol{C}_{V_k}$ obtained by viewpoint selection. We compare a matching score for each pixel in V_k with that for corresponding pixel in $C^n_{V_k} \in \boldsymbol{C}_{V_k}$ $(n = 1, \ldots, N_{pair})$ to keep the consistency among multi-view images. A 3D point $\boldsymbol{M}$ reconstructed from $\boldsymbol{m}$ in V_k and the parameters $\boldsymbol{p}_{V_k}(\boldsymbol{m})$ is transformed into a 3D point $\boldsymbol{M}'$ in the viewpoint $C^n_{V_k}$ by

$$\begin{aligned}\boldsymbol{M}' &= [M'_X, M'_Y, M'_Z]^T \\ &= [\boldsymbol{R}_{C^n_{V_k}} \ \ \boldsymbol{t}_{C^n_{V_k}}]\boldsymbol{R}^{-1}_{V_k}(d_{V_k}(\boldsymbol{m})\boldsymbol{A}^{-1}_{V_k}\tilde{\boldsymbol{m}} - \boldsymbol{t}_{V_k}),\end{aligned} \tag{27}$$

where $\tilde{\boldsymbol{m}}$ is homogeneous coordinates of $\boldsymbol{m}$. A normal vector $\boldsymbol{n}'$ in $C^n_{V_k}$ is defined by

$$\boldsymbol{n}' = [n'_X, n'_Y, n'_Z]^T = \boldsymbol{R}_{C^n_{V_k}}\boldsymbol{R}^{-1}_{V_k}\boldsymbol{n}. \tag{28}$$

Parameters $\boldsymbol{p}'(\boldsymbol{m}')$ in $C^n_{V_k}$ are calculated by

$$\boldsymbol{p}'(\boldsymbol{m}') = \left(M'_Z, \tan^{-1}\left(\frac{n'_X}{n'_Z}\right), \tan^{-1}\left(\frac{n'_Y}{n'_Z}\right)\right). \tag{29}$$

If $Score(V_k, \boldsymbol{p}'(\boldsymbol{m}'), \boldsymbol{m}') > Score(C^n_{V_k}, \boldsymbol{p}_{C^n_{V_k}}(\boldsymbol{m}'), \boldsymbol{m}')$ for the pixel coordinate $\boldsymbol{m}'$ in C_{V_k}, then the depth $d_{C^n_{V_k}}(\boldsymbol{m}')$ and the angle of normal vector $\theta_{C^n_{V_k}}(\boldsymbol{m}')$, $\phi_{C_{V_k}}(\boldsymbol{m}')$ are replaced by $\boldsymbol{p}'(\boldsymbol{m}')$. The above process for all the viewpoints in $\boldsymbol{C}_{V_k}$ provides highly accurate depth and normal estimation, while significantly increasing the computational cost. Therefore, we randomly select only one viewpoint $C^n_{V_k}$ from a set of viewpoints $\boldsymbol{C}_{V_k}$ in view propagation. We found that a limited number of viewpoints can be used to estimate the depth and normal maps with the same accuracy as when the parameters are propagated to all the viewpoints [9]. The above process is performed for all the pixels in V_k.

4.4 Plane refinement

This step is to refine parameters $\boldsymbol{p}_{V_k}$. Increasing the resolution of parameters is necessary to accurately estimate depths and normals. Although the accuracy of parameter estimation can be improved by increasing the resolution of the initial random numbers, it also significantly increases processing time. Plane refinement reduces processing time by refining the parameters by adding random numbers to the parameters with a finer resolution than the resolution of the random numbers generated by the initialization. For a given parameter, we add a random number generated at a finer resolution than the random number used for initialization. Note that one random number is added for each of the parameters. The matching scores are obtained before and after adding the random numbers, and if the addition of a random number increases the score, the parameter is replaced with the parameter to which the random number was added. Thus, for $\boldsymbol{m}$ in V_k, if $Score(V_k, \boldsymbol{C}_{V_k}, \boldsymbol{p}_{V_k}(\boldsymbol{m}) + \Delta\boldsymbol{p}, \boldsymbol{m}) > Score(V_k, \boldsymbol{C}_{V_k}, \boldsymbol{p}_{V_k}(\boldsymbol{m}), \boldsymbol{m})$, the parameter $\boldsymbol{p}_{V_k}(\boldsymbol{m})$ is replaced by $\boldsymbol{p}_{V_k}(\boldsymbol{m}) + \Delta\boldsymbol{p}$, where $\Delta\boldsymbol{p}$ indicates a random number generated for each pixel. In this paper, the range of random numbers is set to 1/4 of the range in initialization described in Sect. 4.1. It is expected to improve the accuracy by performing this process repeatedly. To reduce the processing time, we perform the above procedure three times in one plane refinement in this paper. In addition, the range of $\Delta\boldsymbol{p}$ is reduced by 1/2 for each time.

4.5 3D reconstruction

After repeating spatial propagation, view propagation, and plane refinement N_{itr} times and applying filters to the depth and normal maps, the depth and normal maps for V_k are obtained as shown in Fig. 5. For a viewpoint $V_k \in \boldsymbol{V}$, let the depth of pixel $\boldsymbol{m}$ be $d_{V_k}(\boldsymbol{m})$, the intrinsic parameters be $\boldsymbol{A}_{V_k}$, and the extrinsic parameters be $\boldsymbol{R}_{V_k}$ and $\boldsymbol{t}_{V_k}$. In this case, the 3D point $\boldsymbol{M}$ reconstructed from $\boldsymbol{m}$ is calculated by

$$M = \boldsymbol{R}^{-1}_{V_k}(d_{V_k}(\boldsymbol{m})\boldsymbol{A}^{-1}_{V_k}\tilde{\boldsymbol{m}} - \boldsymbol{t}_{V_k}), \tag{30}$$

where $\boldsymbol{M}$ is the coordinate of a 3D point in the world coordinate system. For every pixel $\boldsymbol{m}$ in viewpoint V_k, we reconstruct a 3D point by Eq. (30). By computing this process for all the viewpoints and integrating the point clouds, we obtain a 3D point cloud that is reconstructed from the input images $\boldsymbol{V}$.

5 Experiments and discussion

In this section, we evaluate the accuracy of the proposed method by using images taken under various conditions.

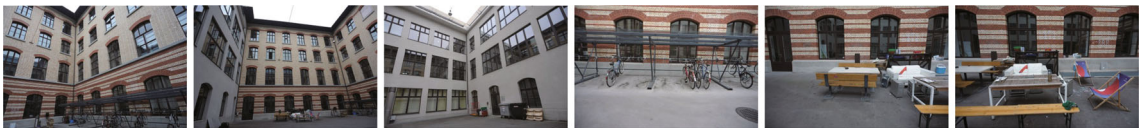

Fig. 7 Example of input images in "courtyard" of the ETH3D dataset

First, we evaluate the effectiveness of each techniques proposed in this paper for PM-MVS through the ablation study. Next, we demonstrate the effectiveness of the proposed method by comparing it with some typical conventional MVS methods using the ETH3D dataset [20] including multi-view images taken in indoor and outdoor environment. Finally, we demonstrate that the proposed method can reconstruct dense and accurate 3D point clouds from multi-view images regardless of the environment and object types through the experiments using the DTU dataset [10] including multi-view images taken in an indoor environment.

5.1 Ablation study

We apply different combinations of the improvements in the proposed method and check their effectiveness. In this experiment, we use "courtyard" in the ETH3D dataset [20]. The "courtyard" set consists of images taken by Nikon D3X from 38 viewpoints. Three types of images are provided in the dataset: RAW images, JPEG images, and distortion-corrected JPEG images. In this experiment, we use only the distortion-corrected JPEG images. Although the image size is approximately $6,048 \times 4,032$ pixels, the image size is reduced by a quarter in this experiment to reduce the processing time. An example of the input image used in the experiment is shown in Fig. 7. For accuracy evaluation, the ETH3D dataset provides a 3D point cloud measured by the FARO Focus X 330 laser scanner. The camera parameters for each viewpoint are provided as the parameters estimated by the SfM tool COLMAP [18] and scaled to match the ground-truth 3D point cloud. In this experiment, the camera parameters for each viewpoint are scaled to fit the image size. In addition, the sparse 3D point cloud reconstructed by SfM of COLMAP is also provided. In the proposed method, a mesh model is generated from this point cloud and used for viewpoint selection.

The parameters of the proposed method used in this experiment are set as follows. We set the matching window size to 10×10 pixels and the number of iterations N_{itr} to 4. The parameters of the viewpoint selection process are set to $\{N_{pair}, N_s, \sigma_s, \tau_{tri}, \sigma_i\} = \{2, 10, 0.6, \pi/180, 45.0\}$. The parameters for BNCC and Geometric Consistency are set to $\{\sigma_f, \sigma_m, \eta, \psi_{max}\} = \{12.0, 3.0, 0.01, 3.0\}$. The parameters of the weighted median filter are set to 11 for the window size, $\sigma_f = 2.0$ and $\sigma_n = 0.6$ for b_i. Only pixels with a matching score greater than 0.5 are reconstructed as having reliable depth and normal. The above settings of parameters in PM-MVS have been experimentally confirmed to be applicable to other datasets as well. We evaluate the proposed method using the quantitative metrics of accuracy, completeness, and F_1-score [20]. "Accuracy" is the ratio of 3D points included in the reconstruction result whose distance to the ground-truth 3D point is less than or equal to the tolerance (tol.). This is a metric that indicates how accurately each point has been reconstructed. "Completeness" is the ratio of ground-truth 3D points whose distance to the reconstruction result is less than or equal to tol. It is a metric that indicates how much of the region has been reconstructed. F_1-score is the harmonic mean of accuracy and completeness, and is a metric that indicates the overall accuracy of the reconstruction result. Since some 3D reconstruction methods have a trade-off between accuracy and completeness, the F_1 score, which is the combination of these two factors, is a good indicator to measure the performance of the methods. The higher the value of each of these metrics, the better the reconstruction result.

Table 1 Specification of the methods compared in the ablation study (VS: viewpoint selection, WM: weighted median filter, GC: geometric consistency)

Method	VS	Matching score	WM
A	View wise	NCC	
B	View wise	BNCC + GC	✓
C	Pixel wise	NCC	✓
D	Pixel wise	BNCC + GC	
E	Pixel wise	BNCC + GC	✓

The methods to be compared in this experiment are summarized in Table 1. A is our previous method [9] with filtering based on the consistency of multi-view geometry [26]. B is a modified version of A with improved matching score calculation and filtering. C is a modified version of A with improved viewpoint selection method and filtering. D is a modified version of A with improved viewpoint selection and matching score calculation. E is the proposed method in this paper with all the improvements.

Table 2 shows the accuracy (A), completeness (C), and F_1-score (F_1) of each method, and Fig. 8 shows the reconstruction results of each method. Note that Fig. 8 shows a magnified view of a part of the reconstruction results. Therefore, there may be a gap between the appearance and the number of reconstructed points in Fig. 8 D compared to other

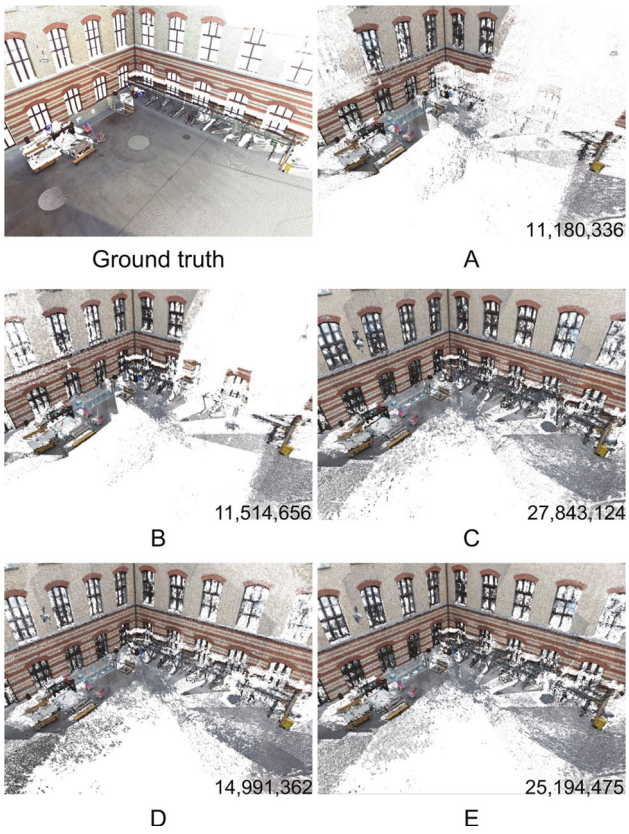

Fig. 8 Reconstruction results of "courtyard" for each method in the ablation study. The number in the each figure indicates the number of reconstructed 3D points

results. In fact, Fig. 8 D has many missing parts that are not shown in the figure, and the upper right corner is sparser than in C and E, resulting in fewer reconstructed points. A and B do not use pixel-wise viewpoint selection, which results in degraded matching accuracy, missing regions on the wall, and the small number of reconstructed 3D points. On the other hand, for C, D, and E, which use pixel-wise viewpoint selection, the missing regions in A and B are recovered and the reconstruction results are dense. Compared with C, which uses only NCC to calculate the matching scores, and D, which does not use a weighted median filter, E shows higher or comparable F_1-scores. The above results show the effectiveness of the proposed method, which employs all the improvement techniques.

5.2 3D reconstruction from multi-view images of ETH3D dataset

We compare the reconstruction accuracy of the proposed method with that of conventional MVS methods through experiments using the ETH3D dataset [20]. The conventional methods are PMVS [4], COLMAP [19], and Yodokawa et al. [26], which is the method A in Table 1 with filtering based on the consistency of multi-view geometry. In this experiment, we use 12 datasets from the training data of High-res multi-view, where we exclude "facade" from this experiment since it has larger number of images then other datasets and some of the methods are out-of-memory. An example of the input images selected from the datasets "delivery area" and "terrace" is shown in Fig. 9. The other experimental conditions are the same as those described in the previous section.

Table 3 shows a summary of experimental results for the ETH3D dataset, where we indicate the results for tol. = 2 cm. COLMAP has the highest accuracy for all the datasets, while the F_1 score is not necessarily high due to the low completeness. The accuracy of the reconstructed 3D points is high, while the range of the reconstructed area is narrow. Yodokawa et al.'s method has a higher completeness than COLMAP on some datasets, although its overall performance is lower than that of COLMAP. The proposed method, PM-MVS, has the highest completeness for all the datasets. The accuracy is lower than COLMAP since the reconstructed area is larger than COLMAP and includes 3D points with lower reconstruction accuracy. While PM-MVS can reconstruct areas with poor texture and far from the camera that cannot be reconstructed by COLMAP, these areas are difficult to be recovered by MVS, resulting in a lower accuracy for PM-MVS. Since there is a trade-off between accuracy and completeness for each method, the F1 score, which is the combination of accuracy and completeness, indicates the performance of each method in MVS. The F_1 score for PM-MVS is the highest in most cases, indicating that the reconstruction effectiveness is high.

We focus on the results of "delivery_area" and "terrace" in the following to analyze the experimental results of each method in detail. Tables 4 and 5 summarize the results for accuracy (A), completeness (C), and F_1-score (F_1) in "delivery_area" and "terrace," respectively. Figures 10 and 11 show reconstruction results of each method in "delivery_area" and "terrace," respectively. The second and third columns of Figs. 10 and 11 show the 3D points colored based on the class of 3D points used in the evaluation of accuracy and completeness, respectively. In accuracy, the reconstructed point cloud is classified into three types: accurate point, inaccurate point, and unobserved point. An accurate point (green) is a point that is accurately reconstructed, an inaccurate point (red) is a point that is inaccurately reconstructed, and an unobserved point (blue) is a point that is not included in the set of ground-truth points. Note that unobserved points are not used for evaluation. More accurate points indicate higher accuracy of the reconstruction. In completeness, ground-truth points are classified into two types: complete points and incomplete points. A complete point (green) is a point where the corresponding 3D point of the reconstruction result exists. An incomplete point (red) is a point where the corresponding 3D point of the reconstruction result does not exist. More complete points indicate higher accuracy of the reconstruction.

Fig. 9 Example of input images in "delivery_area" and "terrace" of the ETH3D dataset

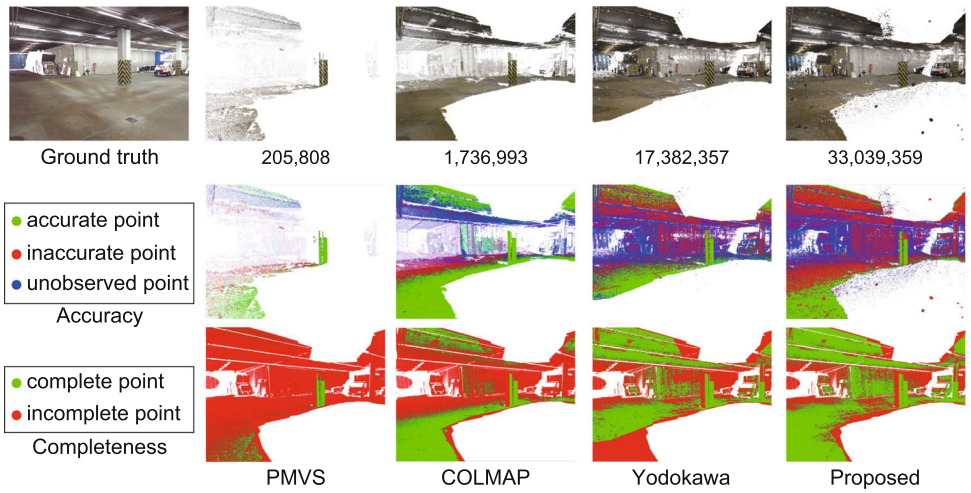

Fig. 10 Results of 3D reconstruction of "delivery_area" (first column: 3D point cloud colored by pixel values, second column: point cloud visualizing accuracy at tol. = 1 cm, and third column: point cloud visualizing completeness at tol. = 1 cm). The number listed below each figure in the first row indicates the number of reconstructed 3D points (color figure online)

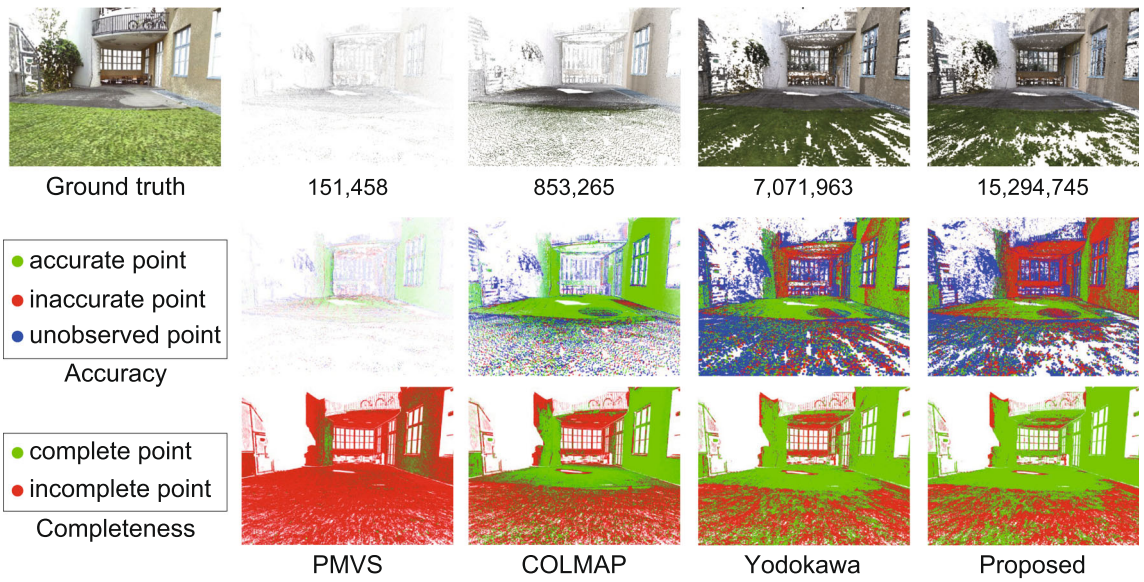

Fig. 11 Results of 3D reconstruction of "terrace" (first column: 3D point cloud colored by pixel values, second column: point cloud visualizing accuracy at tol. = 1 cm, and third column: point cloud visualizing completeness at tol. = 1 cm). The number listed below each figure in the first row indicates the number of reconstructed 3D points (color figure online)

Table 2 Experimental results of "courtyard" for each method in the ablation study: accuracy (A) [%], completeness (C) [%] and F_1-score (F_1) [%] at tol. [cm]

Tol	A			B			C			D			E		
	A	C	F_1	A	C	F_1	A	C	F_1	A	C	F_1	A	C	F_1
1	42.03	14.96	22.07	42.56	17.24	24.54	42.92	**42.21**	**42.56**	**47.53**	30.24	36.96	43.19	40.30	41.69
2	58.83	31.25	40.81	60.01	37.54	46.19	62.82	**72.79**	67.44	**66.43**	65.30	65.86	63.12	72.57	**67.52**
5	79.43	52.21	63.00	80.86	53.01	64.04	84.81	85.21	85.01	**86.63**	83.48	85.03	85.42	**85.45**	**85.44**
10	90.83	67.70	77.58	92.09	57.25	70.61	94.21	88.99	91.52	**95.37**	88.59	91.86	94.77	**89.22**	**91.91**
20	96.61	79.73	87.36	97.42	60.01	74.27	98.07	**91.93**	94.91	**98.67**	91.71	95.06	98.47	91.90	**95.07**
50	98.82	86.28	92.13	99.31	64.35	78.10	99.20	**95.20**	97.16	99.48	95.05	97.22	**99.49**	95.11	**97.25**

A bold font indicates the highest value for each metric

Table 3 Experimental results for the training data of "high-res multi-view" in the ETH3D dataset: accuracy (A) [%], completeness (C) [%] and F_1-score (F_1) [%] for tol. = 2 cm

Dataset	PMVS [4]			COLMAP [19]			Yodokawa [26]			Proposed		
	A	C	F_1	A	C	F_1	A	C	F_1	A	C	F_1
Courtyard	70.18	9.26	16.37	**85.17**	35.93	50.54	60.78	49.78	54.74	63.12	**72.57**	**67.52**
Delivery_area	69.98	16.32	26.47	**90.01**	45.83	60.73	64.97	62.79	63.86	62.38	**76.74**	**68.82**
Electro	68.94	11.61	19.87	**91.48**	43.95	**59.37**	76.63	14.15	23.89	76.32	**45.86**	57.30
Kicker	72.58	22.67	34.55	**91.63**	35.79	51.48	71.32	17.96	28.7	70.83	**52.14**	**60.07**
Meadow	67.36	10.94	18.83	**81.12**	23.27	36.17	60.34	29.30	39.44	59.60	**38.15**	**46.52**
Office	67.68	12.60	21.24	**91.48**	23.79	37.77	63.35	8.21	14.54	65.80	**41.71**	**51.06**
Pipes	80.50	14.83	25.04	**95.54**	19.27	32.07	80.05	8.36	15.15	78.24	**30.50**	**43.89**
Playground	79.11	15.80	26.34	**85.26**	42.62	**56.83**	81.69	20.74	33.08	76.67	**44.93**	56.66
Relief	90.05	26.10	40.47	**95.89**	48.37	64.31	86.39	27.53	41.76	77.59	**74.43**	**75.98**
Relief_2	89.35	25.94	40.21	**94.61**	46.93	62.74	80.47	47.55	59.78	77.06	**73.90**	**75.45**
Terrace	80.87	23.15	35.99	**95.64**	60.30	73.96	86.34	68.95	76.67	80.87	**79.21**	**80.03**
Terrains	86.67	25.59	39.52	**93.42**	52.41	**67.15**	79.93	39.70	53.05	70.48	**63.51**	66.81
Average	76.94	17.90	28.74	**90.94**	39.87	54.43	74.36	32.92	42.06	71.58	**57.80**	**62.51**

A bold font indicates the highest value for each metric

Table 4 Experimental results for "delivery_area": accuracy (A) [%], completeness (C) [%] and F_1-score (F_1) [%] for each method in tol. [cm]

Tol	PMVS [4]			COLMAP [19]			Yodokawa [26]			Proposed		
	A	C	F_1	A	C	F_1	A	C	F_1	A	C	F_1
1	54.46	4.12	7.66	**76.81**	22.4	34.68	46.87	43.72	45.24	43.55	**56.77**	**49.29**
2	69.98	16.32	26.47	**90.01**	45.83	60.73	64.97	62.79	63.86	62.38	**76.74**	**68.82**
5	84.64	45.65	59.31	**96.34**	68.94	80.37	84.97	76.01	80.24	83.88	**89.31**	**86.51**
10	90.79	62.13	73.77	**97.94**	80.94	88.63	93.32	82.04	87.32	93.21	**94.48**	**93.84**
20	94.01	73.57	82.54	**98.53**	89.94	94.04	96.82	85.64	90.89	97.09	**97.09**	**97.09**
50	96.62	85.42	90.68	**99.00**	97.49	98.24	98.30	88.67	93.24	98.67	**98.68**	**98.68**

A bold font indicates the highest value for each metric

The reconstruction results by the proposed method contain more outliers and inaccurate points than those by the conventional methods as shown in Figs. 10 and 11. The proposed method can reconstruct 3D points that are not included in the ground truth, and can also reconstruct areas with poor texture and far from the camera, which cannot be reconstructed by COLMAP and other methods. Therefore, when visualizing the accuracy of the proposed method, there are more inaccurate points and unobserved points than other methods. On the other hand, visualization of the completeness of the proposed method shows that the number of complete points is larger than that of other methods, indicating that the proposed method can reconstruct a dense point cloud. PMVS has relatively high accuracy, while it has the lowest completeness among all the methods. This is because PMVS is based on patch expansion, which makes it difficult to recon-

Table 5 Experimental results for "terrace": accuracy (A) [%], completeness (C) [%] and F_1-score (F_1) [%] for each method in tol. [cm]

Tol	PMVS [4]			COLMAP [19]			Yodokawa [26]			Proposed		
	A	C	F_1	A	C	F_1	A	C	F_1	A	C	F_1
1	65.82	5.29	9.79	**89.01**	32.64	47.77	73.58	48.91	58.76	64.40	**62.62**	**63.50**
2	80.87	23.15	35.99	**95.64**	60.30	73.96	86.34	68.95	76.67	80.87	**79.21**	**80.03**
5	92.23	54.41	68.44	**98.51**	79.20	87.81	94.70	84.52	89.32	93.25	**91.32**	**92.28**
10	95.77	65.57	77.84	**99.12**	86.85	92.58	97.26	91.62	94.36	96.94	**96.74**	**96.84**
20	97.44	72.76	83.31	**99.43**	93.42	96.33	98.38	97.01	97.69	98.51	**99.08**	**98.79**
50	98.95	79.76	88.33	**99.76**	99.00	99.38	99.20	99.59	99.40	99.44	**99.94**	**99.69**

A bold font indicates the highest value for each metric

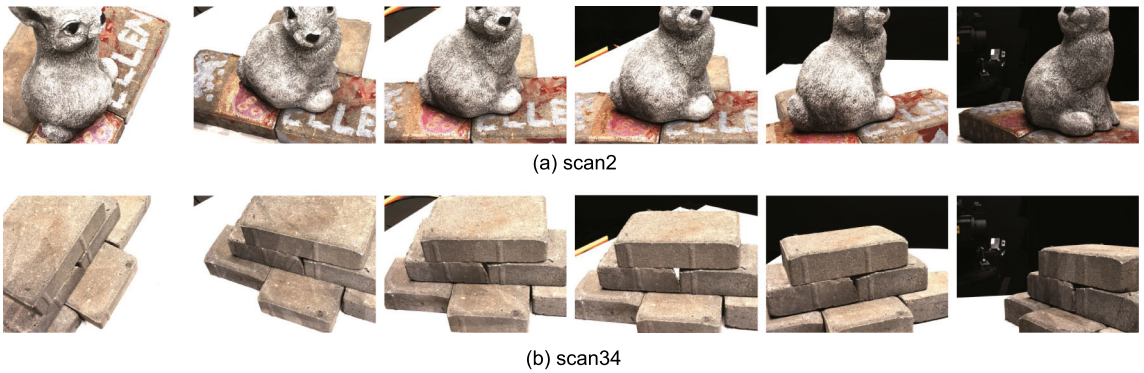

Fig. 12 Examples of input images of the DTU dataset used in the experiment (upper: scan2, lower: scan34)

struct regions with poor texture, such as walls. COLMAP has the highest accuracy, while it has the lower completeness. On the other hand, the proposed method has the highest F_1-score in many tol. and the highest completeness in almost all the cases. As a result, the proposed method can reconstruct the 3D point clouds more densely and accurately than the conventional methods.

5.3 3D reconstruction from multi-View images of DTU dataset

We demonstrate the effectiveness of the proposed method through experiments using the DTU dataset [10]. The DTU dataset provides multi-view images of 128 objects taken under indoor environment, ground-truth 3D point clouds, and camera parameters for each image. The 128 objects include building models, product packages, vegetables, building materials, animal figurines, etc. For each object, the 3D point clouds reconstructed by Campbell et al.'s method [2], PMVS [4], and Tola et al.'s method [23] are also provided. The multi-view images are taken from 49 or 64 viewpoints, and each image has a size of $1,600 \times 1,200$ pixels. In this experiment, we compare the accuracy of the proposed method with that of PMVS [4], COLMAP [19], and Yodokawa et al. [26] as in Sect. 5.2, in addition to the MVS algorithms provided by the dataset. Note that in the multi-view images provided in the DTU dataset, the camera position and pose of the object change automatically, and the light source environment is constant in all the images. Therefore, the proposed method does not use pixel-wise viewpoint selection, but viewpoint selection based on the baseline length as in our previous work [9] in this experiment. In this experiment, we use "scan2" and "scan34" among 128 objects. Both scan sets are taken from 49 viewpoints. Figure 12 shows an example of the input images.

The parameters of the proposed method used in this experiment are almost the same as those in the other experiments, except for the matching window size and N_{pair}. The images in the DTU dataset contain poor-texture regions, therefore, a larger window size and a larger N_{pair} improve the accuracy of the reconstruction. We set the matching window size to 16×16 pixels and the parameters of the viewpoint selection process to $N_{pair} = 4$.

We evaluate the reconstruction accuracy by three metrics: accuracy, completeness, and overall [25], using the evaluation tools provided in the DTU dataset. Note that the definitions of accuracy and completeness are different from those of metrics of the same name in the ETH3D dataset. Accuracy in the DTU dataset is the distance from each point of the reconstructed 3D point cloud to the nearest neighbor point of the ground-truth point cloud. This is a measure of how accurate the reconstructed points are. Completeness in the DTU dataset is the distance from each point in the ground-truth point cloud to the nearest neighbor of the reconstructed

Table 6 Experimental results for "scan2" and "scan34" in the DTU dataset (unit: mm)

	Scan2			Scan34		
	Median acc	Median comp	Median overall	Median acc	Median comp	Median overall
Campbell [2]	0.3740	**0.1463**	0.2601	0.3266	0.1433	0.2350
PMVS [4]	0.2432	0.3088	0.2760	0.2099	0.2890	0.2495
Tola [23]	**0.1807**	0.3432	0.2619	**0.1636**	0.3361	0.2498
COLMAP [19]	0.2058	0.3536	0.2797	0.2013	0.3551	0.2782
Yodokawa [26]	0.2684	0.1754	0.2219	0.1888	**0.1511**	0.1700
Proposed	0.2311	0.1845	**0.2078**	0.1768	0.1532	**0.1650**

A bold font indicates the highest value for each metric

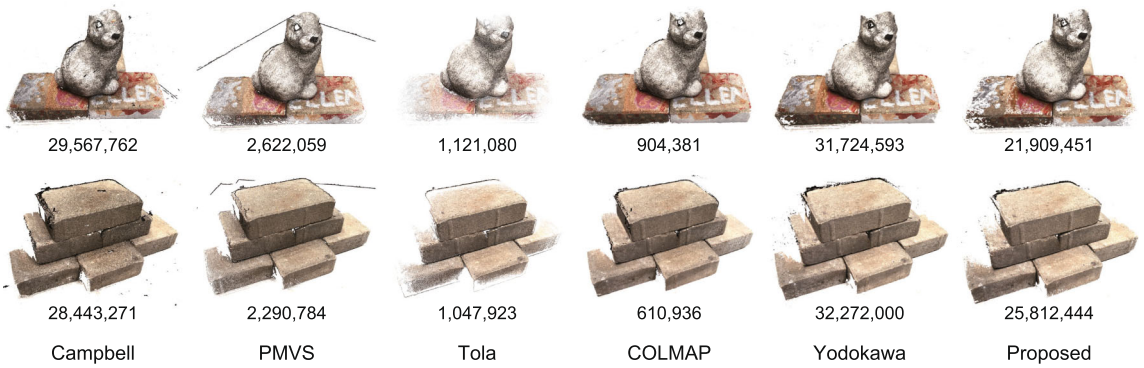

Fig. 13 Reconstruction results of the DTU dataset for each method (upper: scan2, lower: scan34). The number listed below each figure indicates the number of reconstructed 3D points

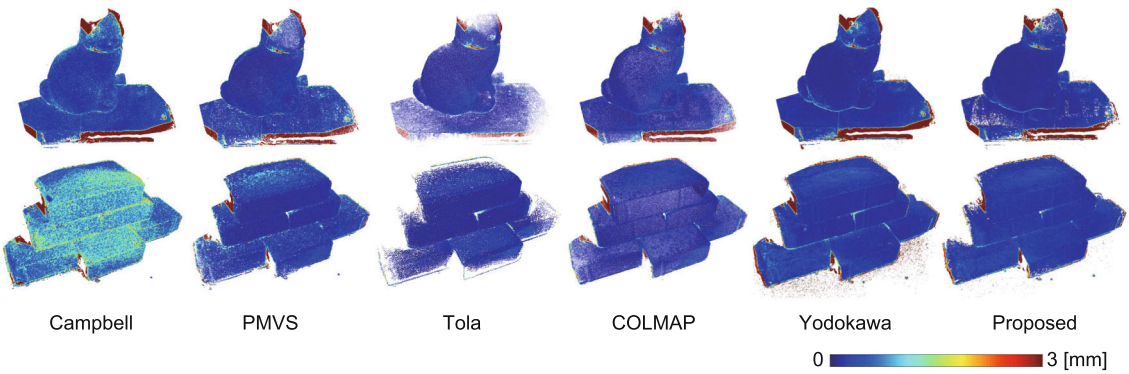

Fig. 14 Error maps of accuracy for each method (upper: scan2, lower: scan34)

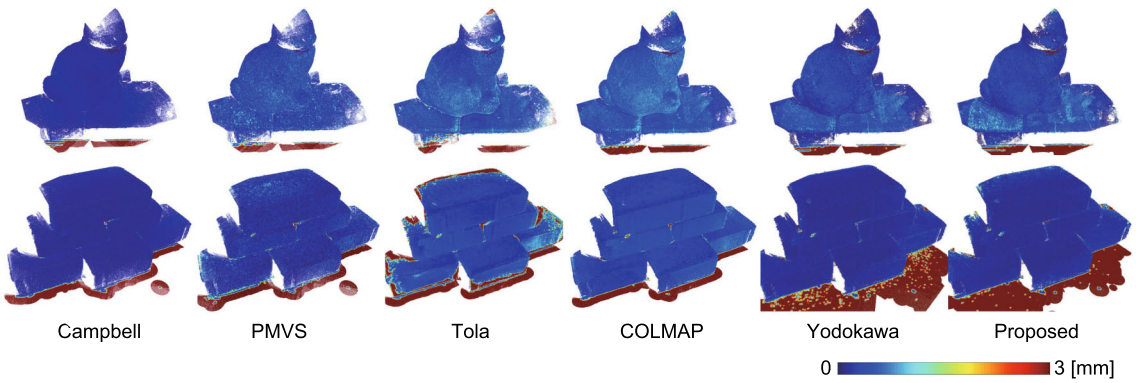

Fig. 15 Error maps of completeness for each method (upper: scan2, lower: scan34)

point cloud. This is a measure of how much of the region of the ground-truth point cloud is reconstructed by the resultant point cloud. Overall was defined as the arithmetic mean of accuracy and completeness, and is a measure of the overall accuracy of the reconstruction results. The lower these metrics are, the higher the accuracy of the reconstruction results.

Table 6 summarizes the median values of accuracy (Acc.), completeness (Comp.), and overall for scan2 and scan34. Figure 13 shows the reconstruction results for each method, Fig. 14 shows error maps of accuracy, and Fig. 15 shows error maps of completeness. In scan2, Acc. of the proposed method is the third lowest after Tola et al. and COLMAP. Comp. is the third lowest after Campbell et al. and Yodokawa et al. On the other hand, overall of the proposed method is the lowest among all the methods. In scan34, Acc. of the proposed method is the second lowest after Tola et al. Comp. is the third lowest after Yodokawa et al.'s method and Campbell et al.'s method. On the other hand, overall of the proposed method is the lowest among all methods as well as scan2. As shown in Figs. 14 and 15, the results of Tola et al.'s method include many points with small Acc. errors, but also many points with large Comp. errors. The results of Campbell et al.'s method include many points with small Comp. errors, but also many points with large Acc. errors. On the other hand, the results of the proposed method include both points with small Acc. and points with small Comp. in a balanced manner, resulting in the most accurate reconstruction results. These results indicate that the proposed method is also effective for 3D reconstruction using data taken in indoor environments.

6 Conclusion

In this paper, we proposed a highly accurate multi-view 3D reconstruction method, PatchMatch Multi-View Stereo (PM-MVS), by introducing three improvement techniques for the extension of PatchMatch Stereo to MVS. In the first technique, the combination of NCC with bilateral weights and geometric consistency between viewpoints was used to improve the estimation accuracy of depth and normal maps at object boundaries and poor-texture regions. In the second technique, the viewpoint to be used for calculating matching scores was selected for each pixel to be robust against disturbances such as occlusion and noise. In the third technique, outliers in the reconstructed 3D point cloud are removed by a weighted median filter and filters based on the consistency of multi-view geometry. Through a set of experiments using public multi-view image datasets, we demonstrated that the proposed method exhibited efficient performance compared with conventional methods. In the future, we will develop a simple and accurate 3D reconstruction system and explore a mesh model generation method using the proposed method.

Acknowledgements The authors would like to thank Mr. Shintaro Ito for his contributions to this research. This work was supported, in part, by JSPS KAKENHI Grant Number 21H03457.

References

1. Bleyer, M., Rhemann, C., Rother, C.: PatchMatch Stereo—Stereo matching with slanted support windows. Proc. Br. Mach. Vis. Confer. **11**, 1–11 (2011)
2. Campbell, N.D.F., Vogiatzis, G., Hernández, C., Cipolla, R.: Using multiple hypotheses to improve depth-maps for multi-view stereo. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) Computer Vision – ECCV 2008. ECCV 2008. Lecture Notes in Computer Science, Vol. 5302. Springer, Berlin, Heidelberg (2008). https://doi.org/10.1007/978-3-540-88682-2_58
3. Tola, E., Lepetit, V., Fua, P.: DAISY: an efficient dense descriptor applied to wide-baseline stereo. IEEE Trans. Pattern Anal. Mach. Intell. **32**(5), 815–830 (2010)
4. Furukawa, Y., Ponce, J.: Accurate, dense, and robust multiview stereopsis. IEEE Trans. Pattern Anal. Mach. Intell. **32**(8), 1362–1376 (2010)
5. Galliani, S., Lasinger, K., Schindler, K.: Massibely parallel multiview stereopsis by surface normal diffusion. Proc. Int'l Confer. Comput. Vis. 873–881 (2015). https://doi.org/10.1109/ICCV.2015.106
6. Goesele, M., Curless, B., Seitz, S.M.: Multi-view stereo revisited. Proc. IEEE Confer. Comput. Vis. Pattern Recogn. 2402–2409 (2006). https://doi.org/10.1109/CVPR.2006.199
7. Goesele, M., Snavely, N., Curless, B., Hoppe, H., Seitz, S.M.: Multi-view stereo for community photo collections. Proc. Int'l Confer. Comput. Vis. (2007). https://doi.org/10.1109/ICCV.2007.4408933
8. Habbeck, M., Kobbelt, L.: A surface-growing approach to multi-view stereo reconstruction. Proc. IEEE Confer. Comput. Vis. Pattern Recogn. 1–8 (2007). https://doi.org/10.1109/CVPR.2007.383195
9. Hiradate, M., Ito, K., Aoki, T., Watanabe, T., Unten, H.: An extension of PatchMatch Stereo for 3D reconstruction from multi-view images. Proc. Asian Confer. Pattern Recogn. 061–065 (2015). https://doi.org/10.1109/ACPR.2015.7486466
10. Jensen, R., Dahl, A., Vogiatzis, G., Tola, E., Anæs, H.: Large scale multi-view stereopsis evaluation. Proc. IEEE Conf. Comput. Vis. Pattern Recogn. 406–413 (2014). https://doi.org/10.1109/CVPR.2014.59
11. Kazhdan, M., Bolitho, M., Hoppe, H.: Poisson surface reconstruction. Proc. Symp. Geom. Process. 61–70 (2006). https://doi.org/10.2312/SGP/SGP06/061-070
12. Lhuillier, M., Quan, L.: A quasi-dense approach to surface reconstruction from uncalibrated images. IEEE Trans. Pattern Anal.

Mach. Intell. **27**(3), 418–433 (2005). https://doi.org/10.1109/TPAMI.2005.44
13. Li, J., Li, E., Chen, Y., Xu, L., Zhang, Y.: Bundle depth-map merging for multi-view stereo. Proc. IEEE Confer. Comput. Vis. Pattern Recogn. 2569–2776 (2010). https://doi.org/10.1109/CVPR.2010.5540004
14. Ma, Z., He, K., Wei, Y., Sun, J., Wu, E.: Constant time weighted median filtering for stereo matching and beyond. Proc. Int'l Confer. Comput. Vis. 49–56 (2013). https://doi.org/10.1109/ICCV.2013.13
15. Romanoni, A., Matteucci, M.: TAPA-MVS: textureless-aware patchmatch multi-view stereo. Proc. Int'l Confer. Comput. Vis. 10413–10422 (2019). https://doi.org/10.1109/ICCV.2019.01051
16. Sakai, S., Ito, K., Aoki, T., Masuda, T., Unten, H.: An efficient image matching method for multi-view stereo. In: Lee, K.M., Matsushita, Y., Rehg, J.M., Hu, Z. (eds.) Computer Vision - ACCV 2012. ACCV 2012. Lecture Notes in Computer Science, Vol. 7727, pp. 283–296. Springer, Berlin, Heidelberg (2013). https://doi.org/10.1007/978-3-642-37447-0_22
17. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. Int. J. Comput. Vis. **47**, 7–42 (2002). https://doi.org/10.1023/A:1014573219977
18. Schönberger, J.L., Frahm, J.M.: Structure-from-motion revisited. Proc. IEEE Confer. Comput. Vis. Pattern Recogn. 4104–4113 (2016). https://doi.org/10.1109/CVPR.2016.445
19. Schönberger, J.L., Zheng, E., Frahm, J.M., Pollefeys, M.: Pixelwise view selection for unstructured multi-view stereo. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) Computer Vision – ECCV 2016. ECCV 2016. Lecture Notes in Computer Science, vol 9907. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46487-9_31
20. 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. Proc. IEEE Confer. Comput. Vis. Pattern Recogn. 2538–2547 (2017). https://doi.org/10.1109/CVPR.2017.272
21. Shen, S.: Accurate multiple view 3D reconstruction using patch-based stereo for large-scale scenes. IEEE Trans. Image Process. **22**(5), 1901–1914 (2013)
22. Szeliski, R.: Computer Vision: Algorithms and Applications. Springer, New York (2010)
23. Tola, E., Strecha, C., Fua, P.: Efficient large-scale multi-view stereo for ultra high-resolution image sets. Mach. Vis. Appl. **23**(5), 903–920 (2012)
24. Xu, Q., Tao, W.: Planar prior assisted PatchMatch multi-view stereo. Proc. AAAI Confer. Artif. Intell. **34**(7), 12516–12523 (2020)
25. Yao, Y., Luo, Z., Li, S., Fang, T., Quan, L.: MVSNet: depth inference for unstructured multi-view stereo. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) Computer Vision – ECCV 2018. ECCV 2018. Lecture Notes in Computer Science, vol 11212. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01237-3_47
26. Yodokawa, K., Ito, K., Aoki, T., Sakai, S., Watanabe, T., Masuda, T.: Outlier and artifact removal filters for multi-view stereo. Proc. Int'l Confer. Image Process. 3638–3642 (2018). https://doi.org/10.1109/ICIP.2018.8451348
27. Zheng, E., Dunn, E., Jojic, V., Frahm, J.M.: Patchmatch based joint view selection and depthmap estimation. Proc. IEEE Confer. Comput. Vis. Pattern Recogn. 1510–1517 (2014). https://doi.org/10.1109/CVPR.2014.196