IEICE TRANS. INF. & SYST., VOL.E98–D, NO.10 OCTOBER 2015

PAPER

Phase-Based Window Matching with Geometric Correction for Multi-View Stereo

Shuji SAKAI[†a)], ***Nonmember***, **Koichi ITO**[†b)], **Takafumi AOKI**[†], ***Members***, **Takafumi WATANABE**[††], ***Nonmember***, ***and*** **Hiroki UNTEN**[††], ***Member***

SUMMARY Methods of window matching to estimate 3D points are the most serious factors affecting the accuracy, robustness, and computational cost of Multi-View Stereo (MVS) algorithms. Most existing MVS algorithms employ window matching based on Normalized Cross-Correlation (NCC) to estimate the depth of a 3D point. NCC-based window matching estimates the displacement between matching windows with sub-pixel accuracy by linear/cubic interpolation, which does not represent accurate sub-pixel values of matching windows. This paper proposes a technique of window matching that is very accurate using Phase-Only Correlation (POC) with geometric correction for MVS. The accurate sub-pixel displacement between two matching windows can be estimated by fitting the analytical correlation peak model of the POC function. The proposed method also corrects the geometric transformations of matching windows by taking into consideration the 3D shape of a target object. The use of the proposed geometric correction approach makes it possible to achieve accurate 3D reconstruction from multi-view images even for images with large transformations. The proposed method demonstrates more accurate 3D reconstruction from multi-view images than the conventional methods in a set of experiments.

key words: *multi-view stereo, window matching, geometric correction, phase-only correlation*

1. Introduction

Multi-View Stereo (MVS) is a technique used to reconstruct the 3D shape of an object using a set of images taken from different viewpoints [1]–[3]. High-quality 3D shapes have recently been created from only camera images with the development of computer and camera technologies and with the advances in 3D reconstruction technologies. Therefore, MVS has attracted considerable attention from various fields such as industry, medical care, and the arts. MVS algorithms consist of combinations of many processes, i.e., selection of views, reconstruction of 3D points by local window matching, removal of outliers, generation of 3D meshes from 3D point clouds, and optimization of 3D meshes. Window matching to determine the 3D coordinates of objects is the most important factor in the processes for the MVS algorithm, since its performance affects the accuracy, robustness, and computational cost of the MVS algorithm.

Manuscript received November 30, 2014.
Manuscript revised April 24, 2015.
Manuscript publicized July 9, 2015.

[†]The authors are with the Graduate School of Information Sciences, Tohoku University, Sendai-shi, 980–8579 Japan.
[††]The authors are with Toppan Printing Co., Ltd., Tokyo, 112–8531 Japan.
a) E-mail: sakai@aoki.ecei.tohoku.ac.jp
b) E-mail: ito@aoki.ecei.tohoku.ac.jp

DOI: 10.1587/transinf.2014EDP7409

Window matching based on Normalized Cross-Correlation (NCC) has been used in most MVS algorithms [1], [4]–[10]. Goesele et al. [4] applied NCC-based window matching to the plane-sweeping approach to generate an accurate depth map by cumulating the correlation values calculated from multiple stereo image pairs with changing depths. Campbell et al. [7] generated a more accurate depth map than that with Goesele et al.'s method [4] by using the matching results from neighboring pixels to improve the accuracy of 3D reconstruction and reduce the number of outliers. Bradley et al. [6] and Furukawa et al. [9] achieved robust window matching by transforming the matching window not only according to depth but also the normal of the 3D points.

NCC-based window matching was used in these MVS algorithms to evaluate the likelihood of 3D points. Therefore, the optimal 3D point has to be found by iteratively computing NCC values with changing parameters of 3D points such as depth and the normal. For instance, the plane-sweeping approach used in Goesele et al.'s algorithm [4] computes NCC values with discretely changing depths of 3D points and selects the depth with the highest NCC value as the optimal one. Since a significantly small step size for depth is required for accurate 3D reconstruction, the number of matches is also significantly increased. In addition, although NCC-based window matching estimates the displacement between matching windows with sub-pixel accuracy by linear/cubic interpolation, such interpolation does not represent accurate sub-pixel values of matching windows.

Addressing the above problems, we proposed an efficient window matching method using Phase-Only Correlation (POC) for MVS [11]. POC (or simply "phase correlation") is a kind of correlation function calculated only from the phase components of 2D Discrete Fourier Transforms (DFTs) of given images [12]–[14]. The sub-pixel displacement between images can be estimated using the analytical peak model of a POC function [14], resulting in accurate depth estimation. However, the accuracy of matching with POC-based method deteriorates in stereo image pairs that have relatively large image transformation, since it is assumed that the image transformation between matching windows only has translational displacement. Although the proposed method demonstrated accurate 3D reconstruction in stereo image pairs with a narrow baseline, the error in reconstruction was increased in stereo image pairs with wide

baselines.

This paper proposes a geometric correction technique to improve the accuracy of the proposed method, where the image transformation between a stereo image pair is approximated by local scaling, skewing, and translations. The matching windows are defined by taking into consideration approximated image transformation. It is important to define the shape of matching windows so as not to change the shape of the POC function to reduce the effect of local scaling and skewing. The proposed method with geometric correction makes it possible to achieve accurate 3D reconstruction from multi-view images. This paper also makes new datasets to evaluate MVS algorithms, which consist of a set of images with camera parameters and their ground-truth data measured by a 3D digitizing system. The proposed approach demonstrated more accurate 3D reconstruction from multi-view images than conventional methods in a set of experiments using public and our own datasets.

The rest of the paper is organized as follows: Section 2 describes the fundamentals of POC for MVS. Section 3 describes the POC-based window matching with geometric correction for MVS. Section 4 demonstrates a set of experiments using public and our own datasets. Section 5 ends with some concluding remarks.

2. Phase-Only Correlation for Multi-View Stereo

This section describes the fundamentals of POC-based window matching for MVS [11]. POC is an image matching technique using the phase components in DFTs of given images and is robust against changes in illumination and noise. Furthermore, the most important feature of POC is that the POC function calculated from two images has an analytical peak model [14]. Translational displacement with sub-pixel accuracy can be estimated by fitting the analytical peak model to the calculated data array around the correlation peak, where the height of the peak and the location of the peak are fitting parameters.

POC is used in local window matching in MVS between multi-view images. Stereo image pairs are generated from multi-view images and then local translational displacement between stereo image pairs is estimated using POC. Since the translational displacement between the stereo image pairs is limited to the direction of epipolar lines, 1D POC-based image matching [15] is used in MVS. The POC functions calculated from stereo images with different viewpoints indicate different peak positions due to the difference in camera positions.

To address the above problem, we introduce the disparity normalization technique to POC-based window matching [11]. Let $\boldsymbol{V} = \{V_0, \cdots, V_{H-1}\}$ be multi-view images with known camera parameters. We consider reference view $V_R \in \boldsymbol{V}$ and neighboring views $\boldsymbol{C} = \{C_0, \cdots, C_{K-1}\} \subseteq \boldsymbol{V} - \{V_R\}$ to be input images, where H is the number of the multi-view images and K is the number of the neighboring views. We generate K pairs of rectified stereo images $V_{R,i}^{\text{rect}}$-C_i^{rect} $(i = 0, \cdots, K-1)$ from V_R and $\boldsymbol{C}$ [1]. The relationship among the 3D point $\boldsymbol{M} = [X, Y, Z]^T$ in the camera coordinate of V_R and the rectified stereo image $V_{R,i}^{\text{rect}}$-C_i^{rect} $(C_i \in \boldsymbol{C})$ with disparity d_i is defined by

$$\boldsymbol{M} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \boldsymbol{R}_i \begin{bmatrix} (u_i - u_{0i})B_i/d_i \\ (v_i - v_{0i})B_i/d_i \\ \beta_i B_i/d_i \end{bmatrix}, \tag{1}$$

where (u_i, v_i) is the corresponding point of $\boldsymbol{M}$ in $V_{R,i}^{\text{rect}}$, (u_{0i}, v_{0i}) is the optical center of $V_{R,i}^{\text{rect}}$, β_i is focal length and B_i is baseline length between $V_{R,i}^{\text{rect}}$-C_i^{rect}. $\boldsymbol{R}_i$ denotes a rotation matrix of the reference view V_R for stereo rectification and is given by

$$\boldsymbol{R}_i = \begin{bmatrix} R_{i11} & R_{i12} & R_{i13} \\ R_{i21} & R_{i22} & R_{i23} \\ R_{i31} & R_{i32} & R_{i33} \end{bmatrix}. \tag{2}$$

The relationship between d_i in each rectified stereo pair and the normalized disparity d can be written as

$$d_i = s_i d, \tag{3}$$

where s_i denotes the scale factor for the disparity d_i and is given by

$$s_i = \frac{(R_{i31}(u_i - u_{0i}) + R_{i32}(v_i - v_{0i}) + R_{i33}\beta_i)B_i}{\frac{1}{K}\sum_{l=0}^{K-1}(R_{l31}(u_l - u_{0l}) + R_{l32}(v_l - v_{0l}) + R_{l33}\beta_l)B_l}. \tag{4}$$

We can integrate the POC functions calculated from multiple stereo image pairs into the same coordinate system by using the normalized disparity d.

We take into consideration the problem of obtaining a true 3D point $\boldsymbol{M}$ from the initial 3D point $\boldsymbol{M}'$ on the reference viewpoint using POC-based window matching in the following. Note that the initial 3D point $\boldsymbol{M}'$ is selected according to the design of the MVS algorithm. Its simplest form is to employ the brute force search to select the initial 3D point $\boldsymbol{M}'$ as used in [4]. Figure 1 overviews depth estimation using POC-based window matching. The initial position of the 3D point $\boldsymbol{M}'$ is projected onto the rectified stereo image pair $V_{R,i}^{\text{rect}}$-C_i^{rect}, and the coordinates on $V_{R,i}^{\text{rect}}$ and C_i^{rect} are denoted by $\boldsymbol{m}_i = [u_i, v_i]^T$ and $\boldsymbol{m}_i' = [u_i', v_i']^T$, respectively. The matching windows f_i and g_i are extracted from $V_{R,i}^{\text{rect}}$ centered at $\boldsymbol{m}_i$ with size $s_i w \times L$ and C_i^{rect} centered at $\boldsymbol{m}_i'$ with size $s_i w \times L$, respectively, where $w \times L$ is the unified size of the matching window. s_i is the scale factor for disparity normalization. Scaling image signals f_i and g_i by $1/s_i$, the size of the matching windows is normalized to $w \times L$, where we denote $\hat{f}_i$ and $\hat{g}_i$ as the scaled version of the matching windows f_i and g_i, respectively. The 1D POC function $\hat{r}_i$ between $\hat{f}_i$ and $\hat{g}_i$ is then calculated. Thus, 1D POC functions $\hat{r}_i$ $(i = 0, \cdots, K-1)$ have the same peak position. The average POC function $\hat{r}_{\text{ave}}$ calculated from POC functions $\hat{r}_i$ $(i = 0, \cdots, K-1)$ is used to improve the accuracy of depth estimation. Note that we average POC functions $\hat{r}_i$ whose peak value α_i is larger than a threshold th_{corr}, to reduce the effects of occlusion and object boundaries. The correlation

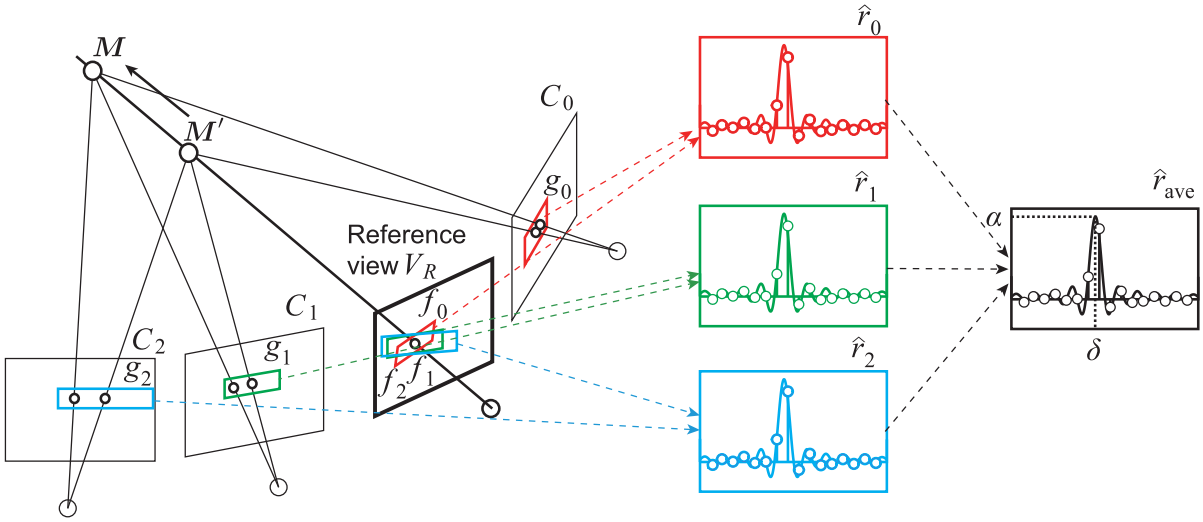

Fig. 1 Depth estimation using POC-based window matching: The initial 3D point $\boldsymbol{M}'$ is projected onto stereo image pairs. POC functions $\hat{r}_i$ are calculated from the matching windows which are extracted from each stereo image pair. The average POC function $\hat{r}_{\text{ave}}$ is calculated from the POC functions. The true 3D point $\boldsymbol{M}$ is obtained by estimating the peak position of $\hat{r}_{\text{ave}}$.

peak position δ with sub-pixel accuracy is estimated by fitting the analytical peak model of the POC function to $\hat{r}_{\text{ave}}$. The true position of the 3D point $\boldsymbol{M}$ is obtained by calculating the displacement between the initial 3D point $\boldsymbol{M}'$ and the true 3D point $\boldsymbol{M}$ from the translational displacement δ as

$$\boldsymbol{M} = \boldsymbol{R}_i \begin{bmatrix} (u_i - u_{0i})B_i/(s_i(d' - \delta)) \\ (v_i - v_{0i})B_i/(s_i(d' - \delta)) \\ \beta_i B_i/(s_i(d' - \delta)) \end{bmatrix}, \tag{5}$$

where d' is normalized disparity of the initial 3D point $\boldsymbol{M}'$. Only one calculation of the POC function makes it possible to calculate the true position of 3D point $\boldsymbol{M}$ with sub-pixel accuracy from the initial 3D point $\boldsymbol{M}'$.

3. POC-Based Window Matching with Geometric Correction for Multi-View Stereo

This section describes POC-based window matching with geometric correction for MVS, which is robust against image transformation between stereo image pairs.

The image transformation between stereo image pairs is represented by nonlinear deformation depending on the 3D shape of the target object and the positions of cameras. Such nonlinear deformation between stereo image pairs is approximated in the proposed method by local scaling, skewing, and translations. POC-based window matching is done between rectified stereo pairs as was explained in Sect. 2. After stereo image pair is rectified, epipolar lines are parallel to the horizontal or vertical axis. Therefore, the image transformation between the rectified stereo image pair is horizontally or vertically limited. Assuming that each local region of the object is approximated by a 3D plane, the image transformation between the matching windows on the rectified reference view $V_{R,i}^{\text{rect}}$ and on the rectified neighboring view C_i^{rect} can be approximated by scaling and skewing as shown in Fig. 2 [16].

The following focuses on the rectified stereo pair $V_{R,i}^{\text{rect}}$-C_i^{rect} to explain how scale factor ξ_i and skew angle κ_i are calculated when given 3D point $\boldsymbol{M}_i = [X_i, Y_i, Z_i]^T$ and its normal vector $\boldsymbol{n}_i = [n_{X,i}, n_{Y,i}, n_{Z,i}]^T$. Note that the coordinate system of $\boldsymbol{M}_i$ and $\boldsymbol{n}_i$, which are the camera coordinates of the rectified reference view $V_{R,i}^{\text{rect}}$, rotates depending on the camera parameter of the neighboring view C_i in stereo rectification. First, we describe the image transformation model between the rectified stereo image pair when assuming that the local region of the object is represented by a 3D plane. Next, we describe the reduction of the effect of scaling and skewing in the matching windows. Then, we present the proposed 3D reconstruction method using POC-based window matching with geometric correction. In the following, we omit the suffix i, which is the index number of stereo pairs, since the scale factor and the skew angle are independently calculated for each stereo pair.

3.1 Binocular Viewing of Plane in Rectified Stereo Pair

At first, we consider that epipolar lines are parallel to the horizontal axis due to stereo rectification as shown in Fig. 2. The rotation matrix $\boldsymbol{R}_{\text{cam}}$ and the translation vector $\boldsymbol{t}_{\text{cam}}$ between the rectified stereo pair are given by

$$\boldsymbol{R}_{\text{cam}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \boldsymbol{t}_{\text{cam}} = \begin{bmatrix} B \\ 0 \\ 0 \end{bmatrix}. \tag{6}$$

The intrinsic parameters $\boldsymbol{A}$ of the reference view V_R^{rect} and $\boldsymbol{A}'$ of the neighboring view C^{rect} are also given by

$$\boldsymbol{A} = \begin{bmatrix} \beta & 0 & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \boldsymbol{A}' = \begin{bmatrix} \beta & 0 & u_0' \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{7}$$

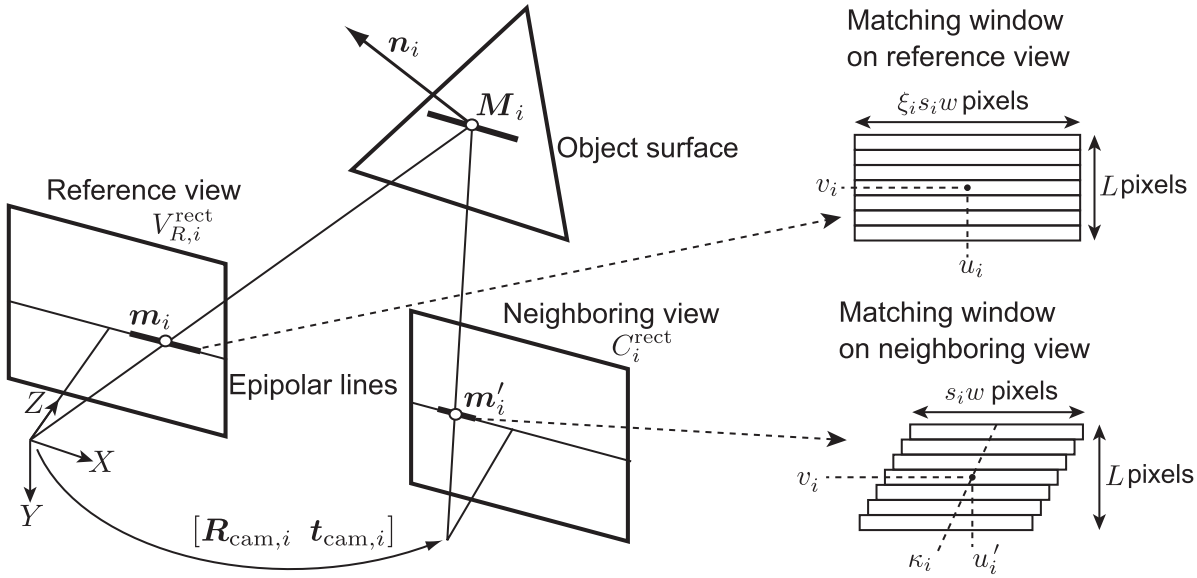

Fig. 2 Matching windows for POC-based window matching with geometric correction for MVS: Assuming that each local region of the object is approximated by a 3D plane, the image transformation between the matching windows can be approximated by scaling and skewing. Local scaling can be reduced by scaling the size of window on the reference view. Local skew can be reduce by skewing the matching window on the neighboring view.

Using the projection from the 3D plane defined by $\boldsymbol{n}$ and $\boldsymbol{M}$, the geometric relation between the coordinate $\boldsymbol{m}$ on the reference view V_R^{rect} and the coordinate $\boldsymbol{m}'$ on the neighboring view C^{rect} is written by

$$s\tilde{\boldsymbol{m}}' = \boldsymbol{H}\tilde{\boldsymbol{m}}. \tag{8}$$

The transformation matrix $\boldsymbol{H}$ is defined by

$$\boldsymbol{H} = \boldsymbol{A}'\left(\boldsymbol{R}_{\mathrm{cam}} + \frac{\boldsymbol{t}_{\mathrm{cam}}\boldsymbol{n}^T}{\boldsymbol{M}\cdot\boldsymbol{n}}\right)\boldsymbol{A}^{-1} \tag{9}$$

$$= \begin{bmatrix} 1 + \frac{Bn_X}{\boldsymbol{M}\cdot\boldsymbol{n}} & \frac{Bn_Y}{\boldsymbol{M}\cdot\boldsymbol{n}} & d_u \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{10}$$

where d_u is given by

$$d_u = \frac{(-u_0 n_X - v_0 n_Y + \beta n_Z)B}{\boldsymbol{M}\cdot\boldsymbol{n}} + u_0' - u_0. \tag{11}$$

As observed in Eq. (10), the transformation matrix $\boldsymbol{H}$ represents an affine transformation between the rectified stereo pair whose target object is a 3D plane. In particular, the transformation matrix $\boldsymbol{H}$ consists of scaling, skewing, and translational displacement in the horizontal axis. In the case that epipolar lines are parallel to the vertical axis, the same discussion can be applied by replacing u and v. As mentioned above, the image transformation between the rectified stereo pair can be represented by scaling, skewing, and translational displacement when it assumes that the local region of the object is approximated by a 3D plane [16].

3.2 Reduction of Effect of Scaling and Skewing in Matching Windows

The key idea of the proposed method is to improve the accuracy of depth estimation of POC-based window matching by reducing the effect of image transformation between matching windows such as scaling and skewing. We do not take into consideration the translational displacement d_u in POC-based window matching, since d_u represents a displacement between the center coordinates of matching windows, which is determined by the initial 3D point $\boldsymbol{M}'$. The matching window has to be defined so as to reduce the effect of scaling and skewing.

In the proposed method, we reduce the effect of image transformation between matching windows by scaling the size of matching window on the reference view by ξ and by skewing the matching window on the neighboring view by κ as shown in Fig. 2. In the case of horizontal stereo rectification, the scale factor ξ and the skew angle κ are given from Eq. (10) as follows:

$$\xi = \left(1 + \frac{Bn_X}{\boldsymbol{M}\cdot\boldsymbol{n}}\right)^{-1}, \quad \kappa = \frac{Bn_Y}{\boldsymbol{M}\cdot\boldsymbol{n}}. \tag{12}$$

Meanwhile, in the case of vertical stereo rectification, ξ and κ are given as follows:

$$\xi = \left(1 + \frac{Bn_Y}{\boldsymbol{M}\cdot\boldsymbol{n}}\right)^{-1}, \quad \kappa = \frac{Bn_X}{\boldsymbol{M}\cdot\boldsymbol{n}}. \tag{13}$$

Local scaling between the stereo image pair can be reduced by scaling the size of matching windows on $V_{R,i}^{\mathrm{rect}}$ into $\xi_i s_i w \times L$ pixels and on C_i^{rect} into $s_i w \times L$ pixels. Note that if the matching window on C_i^{rect} is scaled by $1/\xi_i$ instead of scaling the matching window on $V_{R,i}^{\mathrm{rect}}$, the peak coordinate of $\hat{r}_i$ is not the same as the peak coordinate of $\hat{r}_j$ calculated from a different stereo image pair such as $V_{R,j}^{\mathrm{rect}}$-C_j^{rect}, since the peak coordinate of the POC function $\hat{r}_i$ is scaled by ξ_i.

In addition, the matching window on the neighboring view is transformed by κ_i to reduce the local skew between the stereo image pair, where each vertical line of the matching window is translated on the axis perpendicular to the epipolar line.

3.3 3D Reconstruction Using POC-Based Window Matching with Geometric Correction

We apply the proposed window matching method to a simple plane-sweeping approach to reconstruct 3D point clouds from multi-view images [17] as one of applications of the proposed method. In the basic plane-sweeping approach, the depth of the 3D point is determined by iteratively evaluating a similarity between matching windows with changing depths of the 3D point on the viewing ray on the reference view. As mentioned in Sect. 2, the initial depth is selected according to the design of the MVS algorithm. In this paper, we employ the brute force search to estimate the true depth as well as other window matching method such as [4], i.e., the initial depth is selected from a possible depth range with the step size $\Delta \boldsymbol{Z}$. In addition, the proposed method determines the scale factor and the skew angle between matching windows using the normal vector. Since the normal vectors of the object surface are not known, we select the optimal normal vectors having the highest correlation peak of the POC function by repeating POC-based window matching with changing normal vectors [17]. Given the initial depth of the 3D point, the proposed POC-based window matching method can estimate the true depth with sub-pixel accuracy within the range corresponding to ± 1/4 of the window size w by one window matching. The effective information of POC function with w pixels× L lines is limited to $w/2$ pixels× L lines, since we apply a Hanning widow with $w/2$-half width to the POC function to reduce the boundary effect [15]. Hence, the proposed POC-based window matching method allows us to employ relatively large step sizes within a quarter of the matching window size. On the other hand, NCC-based window matching methods have to employ smaller step size such as $\Delta Z = 1/10$ and $\Delta Z = 1$ than the proposed method in order to reconstruct the accurate 3D points, resulting in the increase in the computational cost. We empirically confirmed that the accuracy of NCC-based window matching methods are significantly dropped when the step size ΔZ is set within a quarter of the matching window size as well as the proposed method.

We calculate a depth on the coordinate $\boldsymbol{m} = [u, v]^T$ in the reference view V_R with the following procedure from all images and camera parameters of the reference view V_R and a set of neighboring views $\boldsymbol{C}$.

Step1: Create a set of rectified stereo image pairs $V_{R,i}^{\text{rect}}$-C_i^{rect} $(i = 0, \cdots, K-1)$ from V_R and $\boldsymbol{C}$.

Step2: Evaluate scale factor s_i for each stereo image pair of image coordinate $\boldsymbol{m}$ on V_R.

Step3: Calculate POC function $\hat{r}_{\text{ave}}$ between the multi-view images with changing the depth and the normal vector $\boldsymbol{n}$ of 3D point $\boldsymbol{M}$, where the matching windows for each stereo pair are transformed by scaling factor ξ_i and skew angle κ_i calculated from $\boldsymbol{n}$ and $\boldsymbol{M}$. We have considered nine candidates for $\boldsymbol{n}$ in this paper, which are obtained by rotating the normal vector facing V_R on the X and Y axes within the range of $\pm\pi/8$. Note that the proposed method does not limit the number of candidates to nine, i.e., this setting is an design example of the proposed method. The number of candidates or the candidate selection method can be designed depending on the MVS algorithm. As mentioned above, the use of the proposed POC-based window matching method makes it possible to employ relatively large step sizes. In this paper, the step size of depth for $\boldsymbol{M}$ corresponds to a quarter of the matching window size on the stereo images.

Step4: Select $\boldsymbol{M}$ and $\boldsymbol{n}$ having the highest correlation peak of $\hat{r}_{\text{ave}}$. Further, update $\boldsymbol{M}$ according to the peak position of $\hat{r}_{\text{ave}}$.

4. Experiments and Discussion

This section describe our evaluation of the accuracy of reconstruction and the computational cost with a variety of window matching methods.

4.1 Dataset for MVS Evaluation

One of the famous datasets for MVS algorithms is the Middlebury MVS dataset [2], [18]. The accuracy of window matching methods cannot be evaluated using this dataset, since this dataset is created for the purpose of evaluating the accuracy of the MVS algorithm including in window matching, view selection, mesh model optimization, etc. On the other hand, the purpose of this paper is to explore the accurate window matching method. Therefore, we do not use the Middlebury MVS dataset to evaluate window matching methods in this paper.

For the purpose of evaluating the accuracy of window matching methods, we make MVS datasets consisting of a set of multi-view images, their camera parameters, and the ground-truth mesh model. Figure 3 shows examples of our MVS dataset. The target objects are figurines of a cat and a dog. We use a camera (Point Gray, Flea 3: FL3-U3-13Y3M-C) with 1,280×1,024 pixels. The images are taken with the camera by changing the height of the camera with 3 patterns and the rotation angle of the turntable with 20 patterns. A 3D mesh model for each target object is measured with the 3D digitizing system (Steinbichler, COMET5) for quantitative performance evaluation. The intrinsic parameters of the camera are estimated using the method of camera calibration proposed by Zhang et al. [19] in advance. The extrinsic parameters are estimated by minimizing the reprojection error of SIFT-based image matching [1], [16], [20], where the reprojection error was calculated using a ground-truth mesh model.

In the experiments, we also employ the dataset "Fountain-P11" [3], [21]. The dataset "Fountain-P11" includes multi-view images (11 images), camera parame-

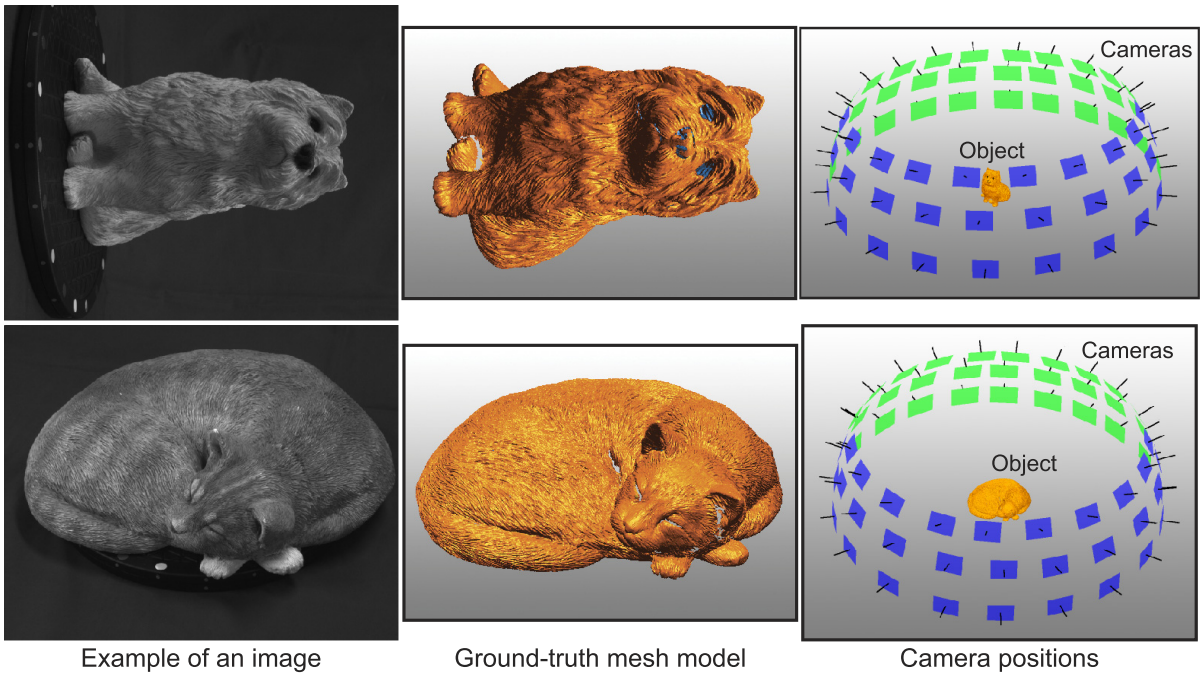

Fig. 3 Example of an image, a ground-truth mesh model, and camera positions in our MVS dataset (Upper: dog, Lower: cat).

ters[†], and a mesh model of a target object, which can be used as the ground-truth. We can evaluate the accuracy of 3D point clouds obtained from window matching methods, since the ground-truth mesh data is publicly available in [21].

4.2 Window Matching Methods

We compare the accuracy of 3D point clouds obtained by window matching methods, since the purpose of this paper is not to propose the whole MVS algorithm but the window matching method for the MVS algorithm. Although some of conventional MVS algorithms employ outlier removal and mesh optimization to improve the accuracy of 3D reconstruction, we evaluate the accuracy of 3D point clouds except for such improvement techniques in the experiments. We classify the window matching methods used in the experiments according to type of matching method, sub-pixel estimation, and geometric correction in Table 1. In the experiments, we apply eight window matching methods to the plane-sweeping approach and evaluate their reconstruction accuracy and computational cost. Both for NCC- and POC-based window matching with geometric correction, we consider nine candidates of the normal vector to estimate a transformation matrix. The following shows detailed description for each method.

NCC+BF (+Homography)

NCC is used for window matching between multi-view images. In BF, NCC values are computed with changing depth Z by the step size ΔZ and the depth with the highest NCC value is selected as the optimal one. In this paper, we set the step size ΔZ corresponding to $1/10$ pixels on the stereo image. The size of window for NCC-based matching is 17×17 pixels, which is equivalent in terms of the effective signal size to that for the POC-based window matching method. The threshold value for averaging the NCC values calculated from stereo image pairs is 0.5. When reducing the effect of image transformation of matching windows in NCC-based window matching methods, the image transformation model is represented by the projective transformation between matching windows. In this case, we do not apply stereo rectification to images.

NCC+FF (+Homography)

In FF, the depth Z with sub-pixel accuracy is estimated by fitting the peak model function around the depth having the maximum NCC value. In this paper, the parabola function is fitted to NCC values around Z having the maximum NCC value obtained by BF with $\Delta Z = 1$ pixel. In this case, we do not apply stereo rectification to images.

NCC+LM (+Homography)

In LM, the depth Z with sub-pixel accuracy is estimated by nonlinear optimization around the depth having the maximum NCC value, where Z is a parameter to be optimized. In this paper, the NCC value is maximized by the Levenberg-Marquardt algorithm, where the initial value is set to the NCC value obtained by BF with $\Delta Z = 1$ pixel. In this case, we do not apply stereo rectification to images.

POC (+Affine)

POC is used for window matching between multi-view

[†]We empirically confirmed that the camera parameters available at [21] had an error. Therefore, we optimized the extrinsic parameters according to the same way used in our datasets.

Table 1 A summary of window matching methods for MVS used in the experiments.

Matching	Sub-pixel estimation	Geometric correction	Reference
NCC	Brute force (BF)	—	Goesele_2006 [4]
		Homography	Bradley_2008 [6]
	Function fitting (FF)	—	
		Homography	
	Nonlinear optimization (LM)	—	
		Homography	Furukawa_2010 [9]
POC	Function fitting (FF)	—	Proposed method
		Affine	

images. The peak values of the POC function are computed with changing depth Z by the step size ΔZ and the depth with the highest peak value is selected as the optimal one. In this paper, we set the step size ΔZ corresponding to a quarter of the matching window size on the stereo images. The threshold value th_{corr} for the peak value of the POC function is 0.5. The size of matching windows $w \times L$ is 32×17 pixels. Note that effective information on the POC function with 32×17 pixels corresponds to information on the matching window with 17×17 pixels of NCC-based window matching, since we apply a Hanning window with $w/2$-half width to the POC function to reduce the boundary effect in DFT computation [15]. When reducing the effect of image transformation of matching windows in POC-based window matching methods, we employ the method described in Sect. 3.3. Note that the combination of stereo rectification and the affine transformation corresponds to the homography transformation used in NCC-based window matching methods.

4.3 Accuracy of 3D Reconstruction

We evaluate the accuracy of 3D point clouds obtained by each method using the error rate between the estimated depth and the true depth. We select 21 images from "dog" and "cat" and 9 images from "Fountain-P11" as the reference view V_R and select 2–4 images in order of distance from each reference view V_R as neighboring views $\boldsymbol{C}$. Note that all the images including in the dataset are used as either a reference view or a neighboring view. We estimate the depths of all the image coordinates on V_R from V_R and $\boldsymbol{C}$ using the window matching methods and evaluate the error rate e defined by

$$e = \frac{|Z_{\text{calculated}} - Z_{\text{ground-truth}}|}{Z_{\text{ground-truth}}}, \tag{14}$$

where $Z_{\text{calculated}}$ is the estimated depth and $Z_{\text{ground-truth}}$ is the true depth on V_R obtained from the ground-truth mesh model. Note that, in the case of "Fountain-P11," we estimate the depth from the pixel coordinates on V_R with the spacing of 4 pixels in order to reduce the processing time. We do not estimate the depth from the area which does not exist ground-truth mesh model on V_R such as background and also do not evaluate matching error on such area.

Figure 4 shows histograms of error rates for each dataset. The histograms are plotted with an interval of 0.01% along the vertical axis. The upper row of Fig. 4 illustrates histograms between error rates and the number of reconstructed 3D points having the associated error rate, while the lower row illustrates histograms between error rates and frequencies of reconstructed 3D points having the associated error rate. Table 2 shows a summary of the total number of reconstructed 3D points and the median of error rates for each dataset. Although we do not apply any outlier removal method to the reconstructed 3D point clouds, the total number of 3D points are different from each method, since we do not estimate the depth on the point whose matching scores are below threshold for all the views. Hence, we observe a different trend between histograms for the number of points and the frequency.

First, we discuss the experimental results using our datasets "dog" and "cat." The 3D point clouds reconstructed by POC+Affine, i.e., our proposed method, have a significant number of points with low error and a low number of points with high error compared with the 3D point clouds reconstructed by other methods. The results for POC indicate the same trend in those for POC+Affine as shown in Fig. 4 and Table 2. The accuracy of 3D points reconstructed by POC is comparable with that by POC+Affine, while the number of 3D points is less than POC+Affine. When image transformation between matching windows is large, the peak value of POC function is significantly dropped. If the peak value is smaller than the threshold th_{corr}, the 3D point is not reconstructed from such matching windows. Therefore, the number of 3D points of POC is less than POC+Affine, since image transformation between matching windows is not corrected in the case of POC.

Next, we discuss the results using "Fountain-P11." The number of 3D points reconstructed by POC and POC+Affine is less than that by NCC-based methods. On the other hand, POC+Affine exhibits good performance compared with other methods, focusing on the histogram of frequencies and the median of error rates. This fact indicates that if the 3D point is reconstructed by the proposed method, its accuracy is significant high. Figure 5 shows reconstructed 3D point clouds and their error maps. Note that outliers including in reconstructed 3D point clouds in Fig. 5 are removed by hand for easy-to-understand illustration. Although there are some points having large error, i.e., red points, regardless of methods, the number of points having low error, i.e., blue points, for POC is more than that for NCC+LM. Similarly, the number of points having

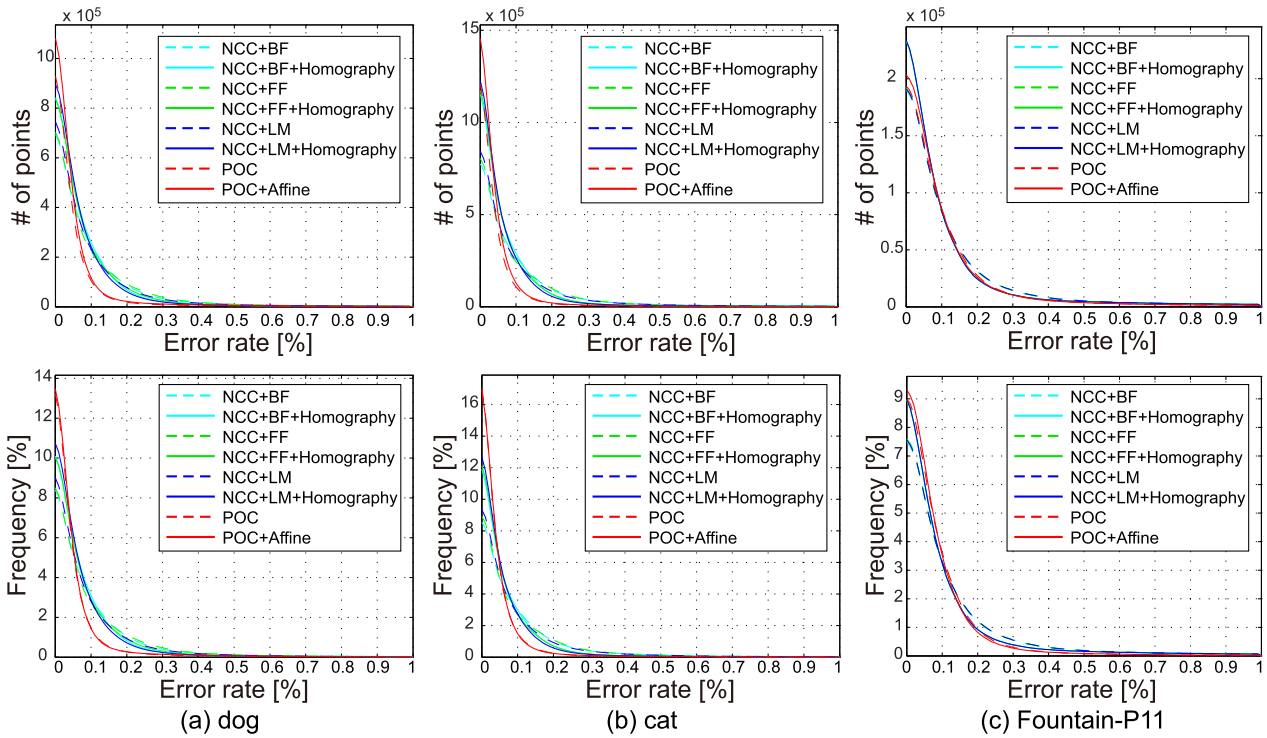

Fig. 4 Histograms of error rates (Upper: Histogram between error rates and the number of reconstructed points, Lower: Histogram between error rates and frequencies of reconstructed points).

Table 2 Summary of experimental results.

	# of points [$\times 10^6$]			Median of error rates [%]		
	dog	cat	Fountain	dog	cat	Fountain
NCC+BF	7.51	8.26	2.53	0.0688	0.0699	0.0804
NCC+FF	7.46	8.21	2.52	0.0684	0.0673	0.0795
NCC+LM	7.46	8.21	2.53	0.0626	0.0629	0.0799
POC	6.15	7.48	2.40	0.0351	0.0310	0.0636
NCC+BF+Homography	7.91	9.30	2.61	0.0564	0.0496	0.0669
NCC+FF+Homography	7.88	9.29	2.60	0.0569	0.0478	0.0671
NCC+LM+Homography	7.88	9.29	2.60	0.0519	0.0454	0.0669
POC+Affine	7.21	9.06	2.45	0.0348	0.0307	0.0604

low error (blue points) for POC+Affine is more than that for NCC+LM+Homography. As observed in the above, the proposed method exhibits more accurate 3D reconstruction than that with all the NCC-based window matching methods.

4.4 Computational Cost

We evaluate the computational cost to estimate the depth of one point on the reference view for each method.

POC-based window matching can estimate the true depth within the range corresponding to ± 1/4 of the window size by one window matching. For NCC-based window matching, we evaluate the computational cost required for depth estimation within the search range equivalent to that in POC-based window matching. The computational cost is evaluated by the number of additions, multiplications, divisions, and square roots required for window matching. Table 3 shows the computational cost to estimate the depth of one point on the reference view using each method. The total cost is calculated as the weighted sum of the number of arithmetic operations, where the weights associated with additions, multiplications, divisions, and square roots are 3, 5, 6, and 6, respectively. We determine the weights based on the latencies of corresponding instructions in Intel®Core™Microarchitecture [22]. As for NCC+LM (+Homography), we evaluate the average computational cost in the Fountain-P11 dataset, since its computational cost depends on the input image. POC (+Affine) includes the computational cost of stereo rectification. The computational cost of stereo rectification is divided by the number of points and added to that of POC (+Affine), since stereo rectification is applied to the whole image once.

Focusing on the type of matching method and subpixel estimation, POC-based methods require lower computational cost than NCC-based methods. By reducing the effect of image transformation, the computational cost in-

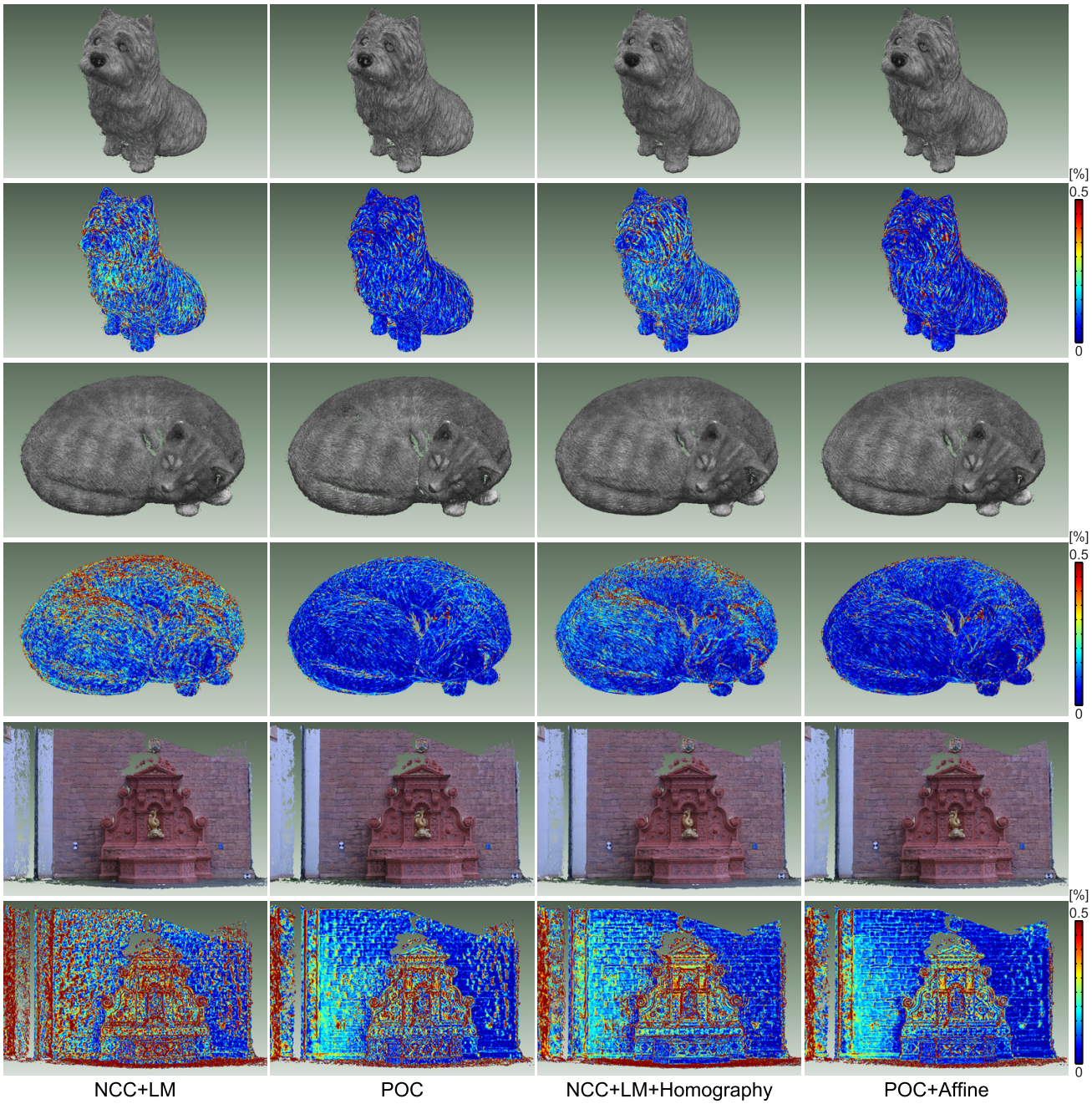

Fig. 5 Reconstructed 3D point clouds and visualized reconstruction error map: "dog" (1~2 rows), "cat" (3~4 rows), and "Fountain-P11" (5~6 rows).

Table 3 Computational cost to estimate the depth of one point on the reference view.

	Additions	Multiplications	Divisions	Square roots	Total cost
NCC+BF	751,400	312,460	5,780	5,780	3,885,860
NCC+FF	75,143	31,250	579	578	388,621
NCC+LM	145,860	60,654	1,122	1,122	754,314
POC	40,060	34,585	2,177	1,088	312,695
NCC+BF+Homography	6,762,600	2,812,140	52,020	52,020	34,972,740
NCC+FF+Homography	676,263	281,218	5,203	5,202	3,497,309
NCC+LM+Homography	740,350	307,865	5,695	5,695	3,828,715
POC+Affine	360,060	310,553	19,585	9,792	2,809,207

creases both for NCC- and POC-based methods. The computational cost for POC+Affine is higher than that for POC, since POC-based matching for nine candidate of the normal vector is required to estimate the optimal normal vector. Although the computational cost for NCC also increases by geometric correction as well as POC, the computational cost for POC+Affine is still lower than that for NCC-based methods with geometric correction.

5. Conclusion

We proposed an efficient method of window matching using POC with geometric correction of matching windows. The proposed approach reduced the nonlinear deformation of matching windows by using scaling and skewing. The method makes it possible to achieve accurate 3D reconstruction even if stereo image pairs have large image transformation. The proposed method demonstrated more accurate 3D reconstruction from multi-view images than conventional window matching methods in a set of experiments. We plan to develop a more efficient MVS algorithm in the future by using the proposed method.

References

[1] R. Szeliski, Computer Vision: Algorithms and Applications, Springer-Verlag New York, 2010.
[2] S.M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski, "A comparison and evaluation of multi-views stereo reconstruction algorithms," Proc. IEEE Conf. Comput. Vis. Pattern Recognit., pp.519–528, June 2006.
[3] C. Strecha, W. von Hansen, L.V. Gool, P. Fua, and U. Thoennessen, "On benchmarking camera calibration and multi-view stereo for high resolution imagery," Proc. IEEE Conf. Comput. Vis. Pattern Recognit., pp.1–8, June 2008.
[4] M. Goesele, B. Curless, and S.M. Seitz, "Multi-view stereo revisited," Proc. IEEE Conf. Comput. Vis. Pattern Recognit., pp.2402–2409, June 2006.
[5] J.P. Pons, R. Keriven, and O. Faugeras, "Multi-view stereo reconstruction and scene flow estimation with a global image-based matching score," Int'l J. Computer Vision, vol.72, no.2, pp.179–193, April 2007.
[6] D. Bradley, T. Boubekeur, and W. Heidrich, "Accurate multi-view reconstruction using robust binocular stereo and surface meshing," Proc. IEEE Conf. Comput. Vis. Pattern Recognit., pp.1–8, June 2008.
[7] N.D.F. Campbell, G. Vogiatzis, C. Hernandez, and R. Cipolla, "Using multiple hypotheses to improve depth-maps for multi-view stereo," Proc. European Conf. Computer Vision, pp.766–779, Oct. 2008.
[8] V.H. Hiep, R. Keriven, P. Labatut, and J.P. Pons, "Towards high-resolution large-scale multi-view stereo," Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp.1430–1437, June 2009.
[9] Y. Furukawa and J. Ponce, "Accurate, dense, and robust multiview stereopsis," IEEE Trans. Pattern Anal. Mach. Intell., vol.32, no.8, pp.1362–1376, Aug. 2010.
[10] A. Delaunoy and E. Prados, "Gradient flows for optimizing triangular mesh-based surfaces: Applications to 3D reconstruction problems dealing with visibility," Int'l J. Computer Vision, vol.95, no.2, pp.100–123, Nov. 2011.
[11] S. Sakai, K. Ito, T. Aoki, T. Masuda, and H. Unten, "An efficient image matching method for multi-view stereo," Proc. Asian Conf. Computer Vision, pp.1–8, Nov. 2012.
[12] C. Kuglin and D. Hines, "The phase correlation image alignment method," Proc. Int'l Conf. Cybernetics and Society, pp.163–165, 1975.
[13] D.J. Fleet, "Phase-based disparity measurement," CVGIP: Image Understanding, vol.53, no.2, pp.198–210, 1991.
[14] K. Takita, T. Aoki, Y. Sasaki, T. Higuchi, and K. Kobayashi, "High-accuracy subpixel image registration based on phase-only correlation," IEICE Trans. Fundamentals, vol.E86-A, no.8, pp.1925–1934, Aug. 2003.
[15] T. Shibahara, T. Aoki, H. Nakajima, and K. Kobayashi, "A sub-pixel stereo correspondence technique based on 1D phase-only correlation," Proc. Int'l Conf. Image Processing, pp.V–221–V–224, Sept. 2007.
[16] R. Hartley and A. Zisserman, Multiple View Geometry, Cambridge University Press, 2004.
[17] D. Gallup, J.M. Frahm, P. Mordohai, Q. Yang, and M. Pollefeys, "Real-time plane-sweeping stereo with multiple sweeping directions," Proc. IEEE Conf. Comput. Vis. Pattern Recognit., pp.1–8, June 2007.
[18] S.M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski, "Multi-view stereo." http://vision.middlebury.edu/mview/
[19] Z. Zhang, "Flexible camera calibration by viewing a plane from unknown orientations," Proc. Int'l Conf. Computer Vision, vol.1, pp.666–673, 1999.
[20] D.G. Lowe, "Distinctive image features from scale-invariant keypoints," Int'l J. Computer Vision, vol.60, no.2, pp.91–110, Nov. 2004.
[21] C. Strecha, "Multi-view evaluation." http://cvlab.epfl.ch/data/
[22] "Intel 64 and IA-32 architectures optimization reference manual." http://www.intel.com/content/dam/doc/manual/64-ia-32-architectures-optimiza tion-manual.pdf

Shuji Sakai received the B.E. degree in information engineering, and the M.S. degree in information sciences from Tohoku University, Sendai, Japan, in 2010 and 2012, respectively. He is currently working toward the Ph.D. degree of the Graduate School of Information Sciences at Tohoku University. His research interest includes signal and image processing, and computer vision.

Koichi Ito received the B.E. degree in electronic engineering, and the M.S. and Ph.D. degree in information sciences from Tohoku University, Sendai, Japan, in 2000, 2002 and 2005, respectively. He is currently an Assistant Professor of the Graduate School of Information Sciences at Tohoku University. From 2004 to 2005, he was a Research Fellow of the Japan Society for the Promotion of Science. His research interest includes signal and image processing, and biometric authentication.

Takafumi Aoki received the BE, ME, and DE degrees in electronic engineering from Tohoku University, Sendai, Japan, in 1988, 1990, and 1992, respectively. He is currently a professor in the Graduate School of Information Sciences (GSIS) at Tohoku University. Since April 2012, he has also served as the Vice President of Tohoku University. His research interests include theoretical aspects of computation, computer design and organization, LSI systems for embedded applications, digital signal processing, computer vision, image processing, biometric authentication, and security issues in computer systems. He received more than 20 academic awards as well as distinguished service awards for his contributions to victim identification in the 2011 Great East Japan Disaster.

Takafumi Watanabe received the B.E. degree in physics from Tokyo University of Science, and the Ph.D. degree in global information and telecommunication from Waseda University, in 2001 and 2007, respectively. He is currently a researcher at Toppan Technical Research Institute in Toppan Printing Co., Ltd. His research interests include virtual reality and computer vision.

Hiroki Unten received the Ph.D. degree in information and communication engineering from the University of Tokyo in 2005. He is currently a researcher at Toppan Technical Research Institute in Toppan Printing Co., Ltd. His research interests include computer vision, 3D measurement, and their applications. He is a member of IEEE and IEICE.