A Sequential Online 3D Reconstruction System Using Dense Stereo Matching

Anonymous WACV submission

Paper ID ****

Abstract

This paper proposes a sequential online 3D reconstruction system using dense stereo matching for a non-expert user, which can sequentially reconstruct accurate and dense 3D point clouds when the new image is captured. The proposed system is based on a novel processing pipeline of sequential online 3D reconstruction with two key techniques: (i) camera parameter estimation of Structure from Motion (SfM) and (ii) dense stereo correspondence matching using Phase-Only Correlation (POC). The user can confirm the reconstruction result and add supplementary images to the system in order to reconstruct a complete 3D model as needed. Through a set of experiments, the proposed system exhibits efficient performance in terms of reconstruction accuracy and computation time compared with the conventional system.

1. Introduction

Recently, with the rapid development of 3D printers, which are now available for consumer use, 3D printing of an object has been receiving much attention. There are still problems for non-experts to utilize the 3D printing technology. One of major problems is to generate a 3D model of a target object in the real-world environment. Most of practical 3D measurement systems employ laser scanning or structured light projection to generate accurate 3D models. These systems require special measurement equipment depending on the target objects and may be liable to be expensive. Also, the application of these systems is limited, since the measurement equipment is relatively large. Hence, an easy-to-use 3D modeling system is indispensable for practical use in daily life.

The topic of reconstructing a 3D model from a set of images has attracted much attention as one of approaches to easy-to-use 3D modeling in the field of computer vision. [20, 22, 5]. A dense and accurate 3D modeling algorithm from still images has been proposed by Furukawa *et al*. [9] and a web-based application of 3D modeling from photos has also been available [2, 1]. Such 3D modeling systems consist of the offline processes, since all the input images are needed to start the 3D modeling process. If the complete 3D model is not generated, the supplementary images have to be taken by the user and all the processes of 3D modeling are performed again. Therefore, the user needs more time to generate a 3D model and the technological knowledge to understand the viewpoint required to generate a complete 3D model.

The 3D modeling algorithm with sequential 3D reconstruction from images is comfortable for non-expert users, since the user easily confirms the 3D reconstruction result and understands the viewpoint required to generate a complete 3D model. So far, online 3D modeling algorithms where the input for these algorithms is a video sequence have been proposed [19, 23, 17, 26, 10, 27]. It is hard for non-expert users to take a high-quality video sequence so as to reconstruct a 3D model of a scene compared with still images, since the image quality of video sequence is lower than that of still images and special hardware devices such as storage, processor, etc. are needed to process a large amount of images. The most difficult problem for non-expert users is to take a video sequence without blur due to hand movement. Recently, the 3D modeling system on mobile phones has been proposed by Tanskanen *et al*. [25]. For the purpose of real-time 3D reconstruction on mobile phones, the system employs Sum of Absolute Differences (SAD) for depth map computation, whose accuracy is not enough to reconstruct an accurate 3D model from images. Also, there is no quantitative evaluation of reconstruction accuracy for the system in [25]. To realize an easy-to-use 3D modeling system, the 3D model has to be sequentially reconstructed from still images, while there is no dense 3D modeling algorithm with sequential 3D reconstruction from still images to the best of our knowledge.

This paper proposes an easy-to-use sequential 3D reconstruction system which combines camera parameter estimation of Structure from Motion (SfM) [24, 11] and dense correspondence matching using Phase-Only Correlation (POC) [21]. When a new image is taken, the proposed system estimates its camera position using SfM and reconstructs dense 3D points using POC from images whose camera position is

known. Through a set of experiments, the proposed system exhibits efficient performance in terms of reconstruction accuracy and computation time compared with the conventional method [9].

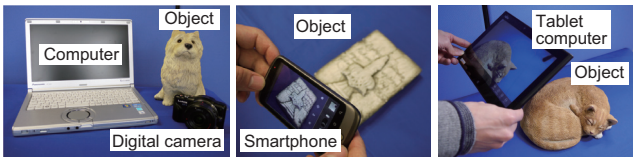
The main features of the proposed system are summarized as follows:

- The effective pipeline for a sequential online 3D reconstruction system is employed, i.e., the proposed system executes 3D reconstruction as well as image acquisition. A user can confirm the reconstruction result and add supplementary images to the system in order to reconstruct a complete 3D model as needed.
- The input for the proposed system is still images, since high-quality images can be taken by a consumer digital camera even for non-expert users.
- The proposed system employs a POC-based correspondence matching method, which allows us to reconstruct a high-quality 3D model.
- The proposed system does not need any special equipment for camera calibration. It is convenient for non-expert users to use the 3D reconstruction system.

2. System Overview

The proposed system reconstructs accurate and dense 3D point clouds from still images taken by a moving camera in a short time and displays the reconstructed 3D point clouds at the same time. The proposed system consists of a consumer digital camera and a computer as shown in Fig. 1. The user can select the type of the system as usage. In the case of Fig. 1 (a), convenience for image acquisition can be improved by using the digital camera or the memory card with Wi-Fi, which transfers images to the computer. In the case of Fig. 1 (b) and (c), the use of a smartphone or a tablet computer makes it possible to realize a portable system, since both a camera and a general-purpose processor are embedded in the system. The input images to 3D reconstruction systems have to be in focus and be without any halation. The images captured with existing consumer digital cameras satisfy the above conditions, since the aperture, shutter speed and focal length of these cameras are automatically configured by their automatic focus and exposure functions. Hence, the user can capture suitable images for 3D reconstruction without any technological knowledge.

The proposed pipeline for the sequential online 3D reconstruction system consists of (i) image acquisition, (ii) pose estimation, and (iii) dense reconstruction as shown in Fig. 2. First, the user captures the image I_i from the arbitrary viewpoint. Second, the system finds correspondence between the image I_i and the previous image I_{i-1} using



(a) (b) (c)

Figure 1. Examples of 3D reconstruction system: (a) consumer digital camera and computer, (b) smartphone, and (c) tablet computer.

feature-based correspondence matching such as SIFT. According to correspondence between I_i and I_{i-1}, the camera parameter of I_i is estimated using SfM [24, 11]. The stereo image pair for the i-th camera position consists of the image I_i and the image $I_j (i > j)$ which is close to I_i. The system rectifies stereo images I_i and I_j and finds the dense correspondence between the rectified stereo images using POC [21]. Then, the system reconstructs dense 3D point clouds from I_i and I_j and merges the reconstructed 3D point clouds to the whole 3D point clouds. The proposed system can simultaneously capture images, update dense 3D point clouds, and display the reconstructed 3D model by using asynchronous multithread processing of the above procedure.

3. Pose Estimation

This section describes details of pose estimation used in the proposed system. The process of pose estimation consists of (i) feature-based correspondence matching, (ii) camera parameter estimation, and (iii) bundle adjustment. Note that this process is executed when more than one image is input, i.e., $i > 1$.

(i) Feature-based correspondence matching

The corresponding point pairs between the images I_i and I_{i-1} are obtained using feature-based correspondence matching, since the stereo images include various geometric transformation such as scaling, rotation, and nonlinear transformation due to a camera movement and a change of focal length. In the proposed system, we employ Scale-Invariant Feature Transform (SIFT) [16]. SIFT is robust against geometric deformation and illumination change between images compared with other feature-based matching methods such as Speeded Up Robust Features (SURF) [6] and Binary Robust Invariant Scalable Keypoints (BRISK) [14]. We also empirically confirm that SIFT exhibits efficient performance of pose estimation in the proposed system compared with other feature-based methods. If the number of corresponding point pairs is not enough to estimate pose estimation, the system requests another image acquisition to the user.

(ii) Camera parameter estimation

The camera parameter of I_i is estimated according to the corresponding point pairs obtained in the previous step.

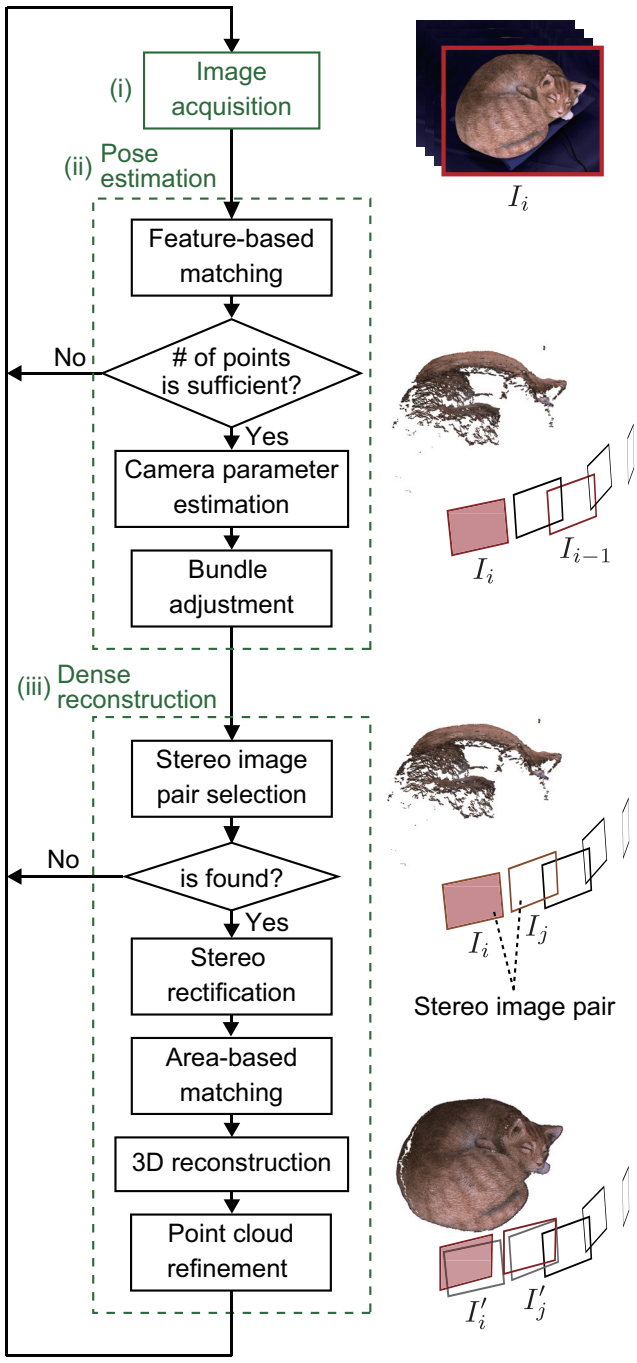


Figure 2. Pipeline of the proposed sequential online 3D reconstruction system: The i-th camera position is estimated from the image I_i taken by the user and the previous image I_{i-1} using SfM. The stereo image pair for the i-th camera consists of the image I_i and the image $I_j (i > j)$ which is close to I_i. We rectify the stereo images and find dense correspondence between the rectified stereo images using POC. We reconstruct 3D point clouds from the corresponding point pairs and update the whole 3D point clouds.

First, we estimate the intrinsic parameter $\boldsymbol{A}_i$ of the camera taking the image I_i. The intrinsic parameter matrix $\boldsymbol{A}$ of a camera is defined by

$$\boldsymbol{A} = \begin{pmatrix} f\frac{w}{D_w} & 0 & \frac{w}{2} \\ 0 & f\frac{h}{D_h} & \frac{h}{2} \\ 0 & 0 & 1 \end{pmatrix}, \tag{1}$$

where f is the focal length of the camera, w and h are the width and height of the image, respectively, and D_w and D_h are the width and height of the image sensor, respectively. The focal length f and the image resolution (w, h) are obtained from Exif (Exchangeable image file format) information of the image [12]. The image sensor size (D_w, D_h) depends on the camera and is given in the specification of the camera. According to Eq. (1), the intrinsic parameter matrix $\boldsymbol{A}_i$ is calculated from I_i.

Next, we estimate the extrinsic camera parameters of I_i. As for $i = 2$, we estimate the extrinsic camera parameters using the normalized five-point algorithm [18]. As for $i > 2$, we estimate the extrinsic camera parameters of I_i using the method proposed by Kneip et al. [13] from the geometric relation between the 3D points and the 2D coordinates of corresponding point pairs. We also employ RANdom SAmple Consensus (RANSAC) [7] for robust parameter estimation.

The 3D points of corresponding point pairs between I_i and I_{i-1} are reconstructed according to the camera parameters and triangulation. We refine reconstructed 3D points using the following three techniques. If the baseline length of the 3D points which have been already reconstructed becomes wide by using the image I_i, we reconstruct the 3D points concerned using I_i again to improve the accuracy of 3D points. We remove a 3D point having too large or too small apical angle between the stereo camera used to reconstruct the 3D point or having too large reprojection error as an outlier. We merge two 3D points whose distance is significant short into one 3D point having the mean coordinate of the original two 3D points to reduce the computation cost of bundle adjustment.

(iii) Bundle adjustment

We optimize the reconstructed 3D points and estimated camera parameters in nonlinear optimization by minimizing reprojection error using bundle adjustment [11, 24], since the accuracy of these parameters has an impact on the succeeding steps. We employ global and local bundle adjustments depending on the target range in this paper.

Global bundle adjustment optimizes all the 3D points and camera parameters of all the images. Let $\boldsymbol{P} = \{\boldsymbol{p}_i\}$ $(1 \leq i \leq K)$ be a set of estimated projection matrices and $\boldsymbol{Q} = \{\boldsymbol{q}_j\}$ $(1 \leq j \leq L)$ be a set of coordinates of a reconstructed 3D point, where K is the number of images and L is the number of 3D points $\boldsymbol{Q}$. Global bundle adjustment

minimizes cost function $E_g(\boldsymbol{P}, \boldsymbol{Q})$ defined by

$$E_g(\boldsymbol{P}, \boldsymbol{Q}) = \frac{1}{2} \sum_{i=1}^{K} \sum_{j=1}^{L} ||\boldsymbol{m}_{i,j} - \boldsymbol{m}_{\mathrm{rep}}(\boldsymbol{p}_i, \boldsymbol{q}_j)||^2, \quad (2)$$

where $\boldsymbol{m}_{i,j}$ is the image coordinate of $\boldsymbol{q}_j$ on the i-th image. The $\boldsymbol{m}_{\mathrm{rep}}(\boldsymbol{p}_i, \boldsymbol{q}_j)$ is the image coordinate reprojected from $\boldsymbol{q}_j$ with $\boldsymbol{p}_i$. Since the computational cost of global bundle adjustment significantly increases with increasing the number of camera parameters and 3D points, we iteratively perform global bundle adjustment at appropriate intervals.

Local bundle adjustment optimizes the camera parameters of the i-th image and 3D points observed in the position of the i-th image. Let $\boldsymbol{p}_i$ be an estimated projection matrix of the i-th image and $\boldsymbol{Q}' = \{\boldsymbol{q}'_j\}$ $(1 \leq j \leq L')$ be a set of coordinates of reconstructed 3D points observed in the position of the i-th image, where L' is the number of 3D points $\boldsymbol{Q}'$. Local bundle adjustment minimizes cost function $E_l(\boldsymbol{p}_i, \boldsymbol{Q}')$ defined by

$$E_l(\boldsymbol{p}_i, \boldsymbol{Q}') = \frac{1}{2} \sum_{j=1}^{L'} ||\boldsymbol{m}_j - \boldsymbol{m}_{\mathrm{rep}}(\boldsymbol{p}_i, \boldsymbol{q}'_j)||^2. \quad (3)$$

Since the computational cost of local bundle adjustment is low, we iteratively perform local bundle adjustment after estimating the camera parameters. We employ the Levenberg-Marquardt (LM) method, which is one of the nonlinear least-squares optimization methods, in these nonlinear optimization processes.

The optimized reconstructed 3D points and estimated camera position and poses are displayed when the pose estimation process is finished.

4. Dense Reconstruction

This section describes dense 3D reconstruction from images whose position is obtained in pose estimation. The process of dense reconstruction consists of (i) stereo image pair selection, (ii) stereo rectification, (iii) dense stereo matching and 3D reconstruction, and (iv) point clouds refinement.

(i) Stereo image pair selection

Dense 3D reconstruction is executed between the stereo image pair having small perspective distortion so as to prevent the quality of 3D reconstruction from decreasing by dense correspondence matching. We select one image $I_j (j < i)$ whose position is closest to the image I_i and which is captured with the parallel optical axis, according to the following procedure. The adjacency of I_n against I_i is defined by

$$\theta_n = (\boldsymbol{r}_i \cdot \boldsymbol{r}_n)(\boldsymbol{d}_i \cdot \boldsymbol{d}_n), \quad (4)$$

where $\boldsymbol{d_n}$ and $\boldsymbol{r_n}$ correspond to the unit vector along the optical axis of the image and the unit vector in the direction from the image to the centroid of the 3D points reconstructed using SfM, respectively. We select the image I_j satisfying the following condition:

$$\theta_{\mathrm{lower}} < \theta_j < \theta_{\mathrm{upper}}, \quad (5)$$

where θ_{lower} and θ_{upper} indicate the lower bound and upper bound of the nearness, respectively. If there is no image satisfying Eq. (5), the dense reconstruction process is not executed for the image I_i.

(ii) Stereo rectification

To obtain accurate and dense 3D points, we employ an area-based correspondence matching method. However, it is hard for an area-based method to obtain correspondence between the stereo image pair having large perspective distortion. Hence, we reduce the perspective distortion between the stereo image pair I_i and I_j by stereo rectification. Stereo rectification is to transform an image pair as if the image pair is captured with a parallel stereo camera [24], that is, the scaling in vertical direction and rotation between the image pair are reduced and the geometric deformation between the image pair is also limited to horizontal direction. Note that the correspondence search between a stereo image pair is reduced to 1D search by stereo rectification. The rectified stereo image pair I'_i and I'_j is obtained by transforming I_i and I_j with the homography matrix calculated from the camera parameters.

(iii) Dense stereo matching and 3D reconstruction

We obtain dense corresponding point pairs between rectified stereo images I'_i and I'_j using an area-based correspondence matching method. Unlike the feature-based correspondence matching method, the area-based correspondence matching method can obtain the point on I'_j corresponding to the reference point placed on the arbitrary position in I'_i. Hence, when many reference points are placed on I'_i, the dense corresponding points can be obtained so as to measure the fine 3D structure of the object.

Among area-based correspondence matching methods, the proposed system employs POC [21], since it can estimate translational displacement between images with subpixel accuracy and is robust against illumination change between images. This advantage is suitable for change of lighting condition and brightness change due to automatic gain selection of the camera. The use of POC makes it possible to reconstruct the accurate 3D shape of the object, since coordinates of corresponding point pairs are with subpixel accuracy according to the analytical correlation peak model of POC function. POC also provides the matching score between local image blocks from the maximum correlation peak value of the POC function.

We reconstruct dense 3D points from corresponding point pairs between I_i and I_j according to triangulation.

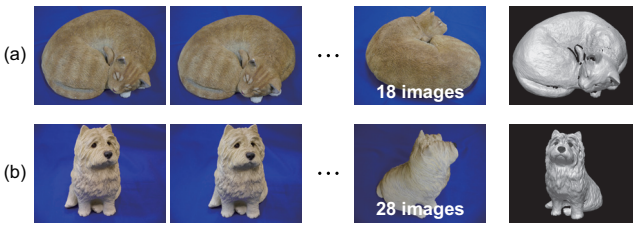


Figure 3. Data sets and ground truth data used in the experiments: (a) Cat and (b) Dog

We remove 3D points having large variance of the distance among nearest K-3D points as an outlier.

(iv) Point clouds refinement

The 3D point clouds $\boldsymbol{C}_i$ reconstructed in the previous step are merged into the 3D point clouds $\boldsymbol{C}$ which have been already reconstructed. Both 3D point clouds $\boldsymbol{C}_i$ and $\boldsymbol{C}$ are represented by octrees, where the smallest size of the octree cell is the mode of the distance between the nearest-neighbor points. For each octree cell, only 3D points having the maximum matching score calculated in dense stereo matching using POC are held and others are removed. We calculate the coordinates of the whole 3D point clouds $\boldsymbol{C}$, if the camera parameters estimated before the image index i are updated by global bundle adjustment in pose estimation. The updated whole 3D point cloud is displayed when the dense reconstruction process is finished.

5. Experiments and Discussion

This section describes a set of experiments to evaluate performance of the proposed system using data sets taken by a moving camera. The performance of the proposed system is compared with the conventional system which consists of (i) camera parameter estimation using SIFT-based SfM and (ii) dense reconstruction using the Patch-based Multi-View Stereo (PMVS) algorithm proposed by Furukawa *et al*. [9].

5.1. Experimental Condition

The target objects are figurines of a cat with W30cm×D30cm×H10cm and a dog with W20cm×D15cm×H20cm as shown in Fig. 3. We use a consumer digital camera (Panasonic LUMIX DMC-GF6) with $1,280 \times 960$ color pixels. We use 18 images for the cat and 28 for the dog. These data sets are taken in advance, since the conventional system is based on the offline process. The camera and the target object are about 1m apart. The initial values of the intrinsic camera parameters are estimated using Exif information of captured images. A 3D mesh model for each target object is measured with the laser scanner (Steinbichler COMET 5) as shown in Fig. 3 to quantitatively evaluate the performance.

3D point clouds are reconstructed from the captured images using the conventional and proposed systems. We assume that the image acquisition time per image is 2 seconds. As for the proposed system, the images are input with 2-second interval. As for the conventional system, the total time of image acquisition is added to the processing time, since the conventional system is not online. We can create a mask to separate the background regions from the image by simple subtraction, since the images are acquired against a blue background. The accuracy of reconstruction is evaluated by comparing the reconstructed 3D point clouds and the ground-truth mesh model using the Iterative Closest Point (ICP). Note that not only rotation and translations but also scale are estimated with ICP, since the scale of the reconstructed 3D point clouds is indefinite.

We implement the conventional system using Visual SfM [28] to estimate the camera parameters and CMVS2 [8] for dense reconstruction. The input images are all the images, where the blue background is masked by black. The level of image pyramid for CMVS2 is 0, the size of matching window is 7×7 pixels, the threshold of the correlation value is 0.6, and the size of the cells is 2×2. The other parameters are the same as those in Furukawa et al. [9].

We implement the proposed system using C++. SIFT-based correspondence matching is implemented using OpenCV [3]. The threshold of reprojection error for RANSAC is 0.5 pixels and the maximum number of iterations for RASAC is 100. The threshold of reprojection error and the minimum apical angle for outlier removal in SfM are 1.0 pixels and 3 degrees, respectively. Global and local bundle adjustment are implemented using Sparse Bundle Adjustment (SBA) [15]. Note that global bundle adjustment is executed when RMS of the reprojection errors of all the camera is more than 0.5 pixels or RMS of the reprojection error is below 0.5 pixels for the 5th time in a row to frequently execute global bundle adjustment. The threshold for stereo image selection is 0.97 for θ_{lower} and 0.99 for θ_{upper}. The interval of reference points placed on the image is 2 pixels. The size of window for POC-based image matching is 32×32 pixels, and the threshold of the peak value is 0.6. We use Point Cloud Library (PCL) [4] to display the reconstructed 3D point clouds and the estimated camera positions, remove outliers of 3D point clouds using a statistical outlier removal filter, and merge 3D point clouds using octree. For outliner removal using the statistical filter, the number of the nearest-neighbor points is 30 and the threshold of the distance variance is 2. Both systems are implemented on Windows 7 Professional, Intel® Core™ i7-990X (3.47GHz), RAM 24 GB.

5.2. Experimental Results

Fig. 4 shows the reconstructed 3D point clouds and error maps for the conventional and proposed systems. "Er-

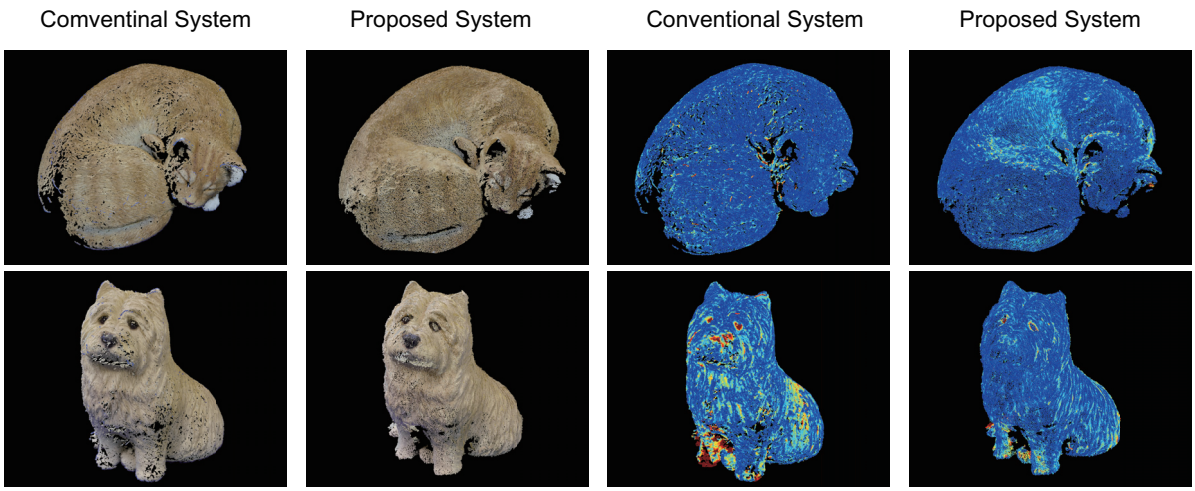


Figure 4. Reconstructed 3D point clouds (1–2 columns) and error maps (3–4 columns) whose range is from blue (0 mm) to red (3 mm).

Table 1. Summary of experimental results. The error in the bracket indicates the error calculated using 3D points that have error below 3 mm. The number of points in the bracket indicates the rate of 3D points that have error below 3 mm.

Data set	Conventional system			Proposed system		
	Error [mm]	# of points	Time [sec.]	Error [mm]	# of points	Time [sec.]
Cat	0.44 (0.43)	334,589 (99.97%)	175	0.47 (0.47)	345,433 (99.98%)	43
Dog	19.89 (0.87)	203,355 (98.11%)	173	0.67 (0.64)	188,904 (99.69%)	52

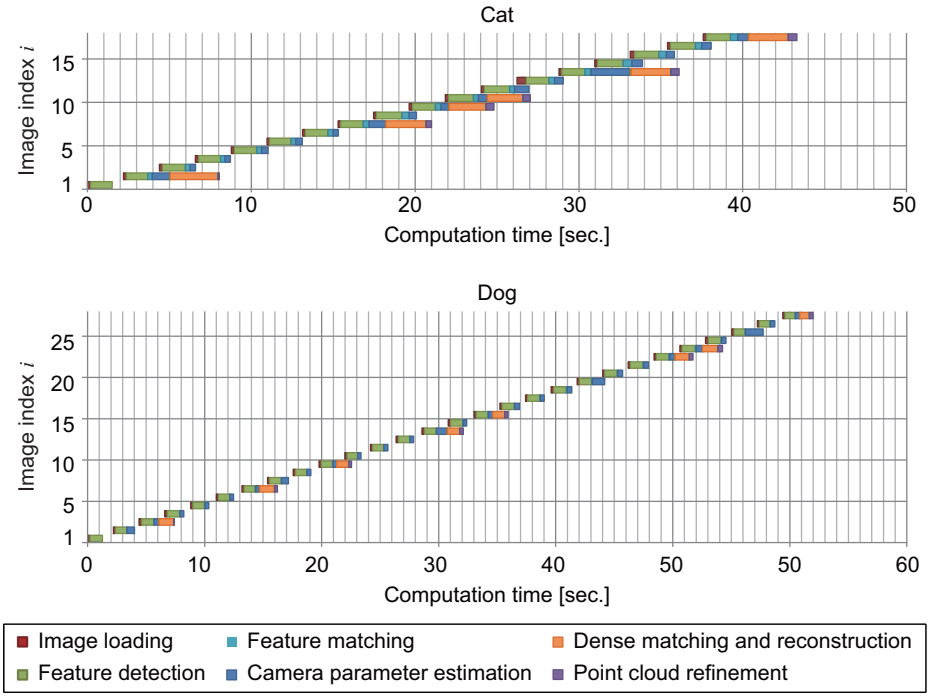


Figure 5. Detailed computation time for each process in the proposed system.

ror" in Table 1 summarizes the Root Mean Square (RMS) of reconstruction errors. The reconstruction accuracy of the proposed system is better than that of the conventional system. Hence, the quality of 3D point clouds reconstructed by the proposed system is sufficiently-high, since the 3D reconstruction method used in the conventional system is

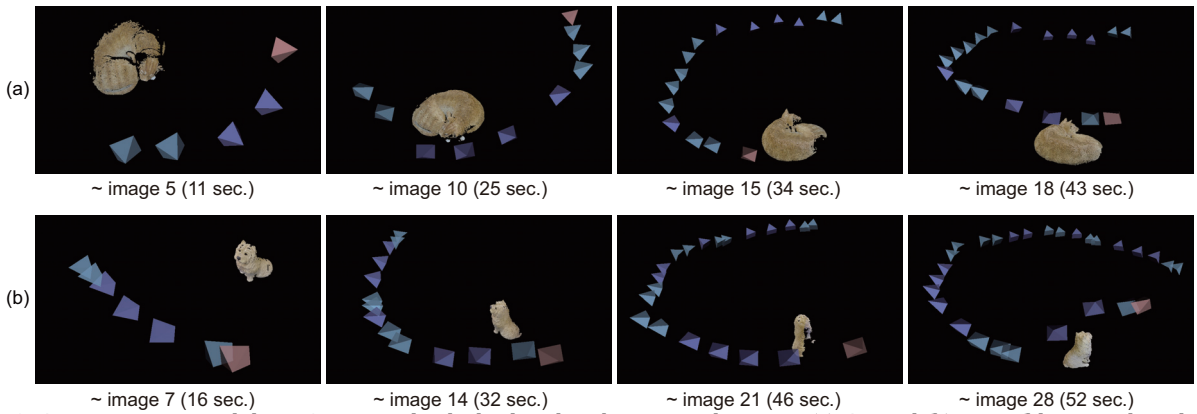


Figure 6. Camera position and dense 3D point clouds displayed in the proposed system: (a) Cat and (b) Dog. (blue: unselected camera, cyan: selected camera, red: the final camera position).

known as one of the accurate ones. From the results of the proposed system as shown in Fig. 4, reconstruction error is increased around the intersection of 3D points reconstructed in the first and last time. Multiple 3D point clouds may not be completely overlapped, since accumulative error of camera pose estimation is increased with increasing the number of images. To address the above problem, we can add loop closing in pose estimation for robust parameter estimation and ICP in the post processing for accurate 3D reconstruction of the whole shape of the object. "# of points" in Table 1 shows the number of reconstructed 3D points. The proposed system can reconstruct a number of 3D points comparable with the conventional system.

"Time" in Table 1 shows the computation time to reconstruct the whole 3D shape for each object. Fig. 5 shows detailed computation time for each process in the proposed system. The computation time of the proposed system is about one-third of that of the conventional system to reconstruct the whole 3D shape of the object. When a new image is input to the system, the conventional system has to do dense reconstruction using PMVS again, in other words, it takes additional hundreds seconds for the conventional system. On the other hand, the proposed system takes a few seconds to do dense reconstruction as shown in Fig. 5, since the proposed system hides the processing time of computationally expensive processes such as SIFT-based feature matching and dense 3D reconstruction using multithread processing. Hence, the proposed system can provide an interactive interface for users through a process of 3D modeling as shown in Fig. 6.

Fig. 7 shows other interesting examples of 3D point clouds reconstructed from images taken by the consumer digital camera. Settings are the same in the above experiments both for conventional and proposed systems. The proposed system can reconstruct dense and accurate 3D point clouds of all the objects.

As observed in the above experiments, the proposed system exhibits efficient performance compared with the conventional system. Also, the use of the proposed system makes it possible to realize online 3D modeling for non-expert users.

6. Conclusion

This paper has proposed a sequential online 3D reconstruction system using dense stereo matching, which can sequentially reconstruct 3D point clouds when a new image is taken. Through a set of experiments, we have demonstrated that the proposed system exhibits efficient performance in terms of reconstruction accuracy and computation time compared with the conventional system. In future work, we will develop an interactive interface so as to support image acquisition by users. Also, we will reconstruct large-scale 3D models of a variety of scenes using the proposed system.

References

[1] Agisoft PhotoScan — www.agisoft.ru. `http://www.agisoft.ru/products/photoscan`.

[2] Autodesk 123D Catch — 3D model from photos. `http://www.123dapp.com/catch`.

[3] Open Computer Vision Library. `http://sourceforge.net/projects/opencvlibrary/`.

[4] Point Cloud Library. `http://pointclouds.org/`.

[5] S. Agarwal, N. Snavely, I. Simon, S. M. Seitz, and R. Szeliski. Building Rome in a day. *Proc. Int'l Conf. Computer Vision*, pages 72–79, Oct. 2009.

[6] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-up robust features (SURF). *Computer Vision and Image Understanding*, 110(3):346–359, 2008.

[7] M. A. Fischler and R. C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Comm. ACM*, 24(6):381–395, 1981.

[8] Y. Furukawa, B. Curless, S. M. Seitz, and R. Szeliski. Towards internet-scale multi-view stereo. *Proc. Int'l Conf.*

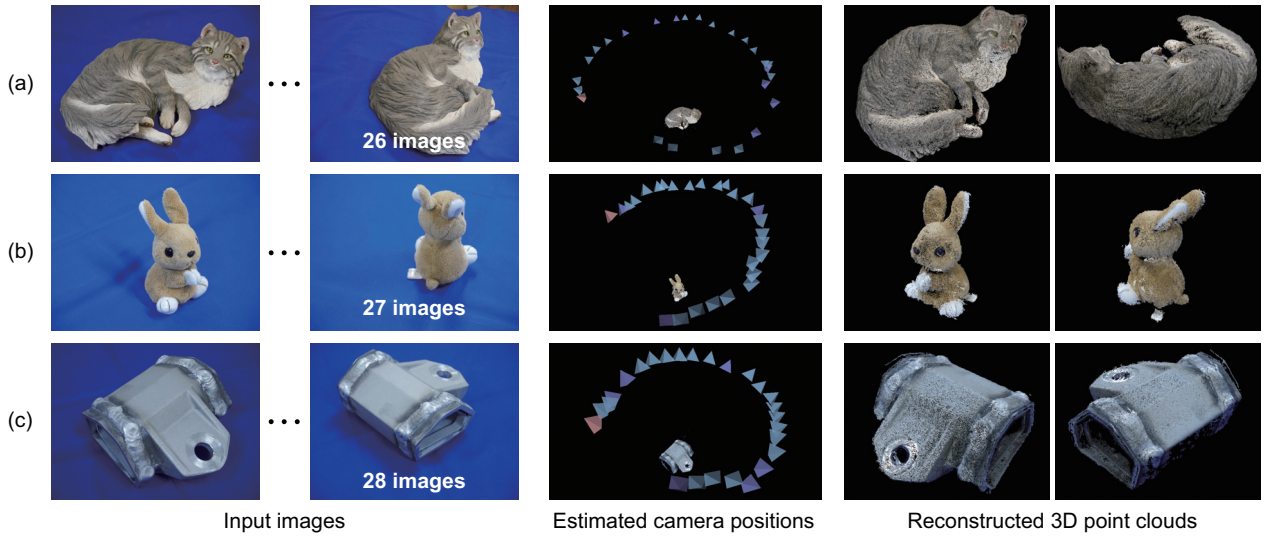


Figure 7. Reconstruction results from images taken by a consumer digital camera: (a) Cat with 26 images, (b) Rabbit with 27 images and (d) metal component with 28 images (First column: input images, 2–3 column: reconstructed 3D point clouds using the proposed system).

Computer Vision and Pattern Recognition, pages 1434–1441, June 2010.

[9] Y. Furukawa and J. Ponce. Accurate, dense, and robust multiview stereopsis. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 32(8):1362–1376, Aug. 2010.

[10] G. Graber, T. Pock, and H. Bischof. Online 3d reconstruction using convex optimization. In *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, pages 708–711. IEEE, 2011.

[11] R. Hartley and A. Zisserman. *Multiple View Geometry*. Cambridge University Press, 2004.

[12] Japan Electronics and Information Technology Industries Association. Exchangeable image file format for digital still cameras. `http://www.jeita.or.jp/`.

[13] L. Kneip, D. Scaramuzza, and R. Siegwart. A novel parametrization of the perspective-three-point problem for a direct computation of absolute camera position and orientation. pages 2969–2976. IEEE, 2011.

[14] S. Leutenegger, M. Chli, and R. Y. Siegwart. Brisk: Binary robust invariant scalable keypoints. pages 2548–2555. IEEE, 2011.

[15] M. I. A. Lourakis and A. A. Argyros. SBA: A software package for generic sparse bundle adjustment. *ACM Trans. Math. Software*, 36(1):1–30, Mar. 2009.

[16] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int'l J. Comput. Vision*, 60(2):91–110, Nov. 2004.

[17] R. Newcombe. DTAM: Dense tracking and mapping in real-time. *Proc. Int'l Conf. Computer Vision*, 2011.

[18] D. Nistér. An efficient solution to the five-point relative pose problem. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 26(6):756–770, 2004.

[19] M. Pollefeys, D. Nistér, J. M. Frahm, A. Akbarzadeh, P. Mordohai, B. Clipp, C. Engels, D. Gallup, S. J. Kim, P. Merrell, C. others Salmi, S. Sinha, B. Talton, L. Wang, Q. Yang, H. Stewénius, R. Yang, G. Welch, and H. Towles. Detailed real-time urban 3d reconstruction from video. *International Journal of Computer Vision*, 78(2-3):143–167, 2008.

[20] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-views stereo reconstruction algorithms. *Proc. Int'l Conf. Computer Vision and Pattern Recognition*, pages 519–528, June 2006.

[21] 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*, 5:V–221–V–224, 2007.

[22] 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. Int'l Conf. Computer Vision and Pattern Recognition*, pages 1–8, June 2008.

[23] J. Stühmer, S. Gumhold, and D. Cremers. Real-time dense geometry from a handheld camera. In *Pattern Recognition*, pages 11–20. Springer, 2010.

[24] R. Szeliski. *Computer Vision: Algorithms and Applications*. Springer-Verlag New York Inc., 2010.

[25] P. Tanskanen, K. Kolev, L. Meier, F. Camposeco, O. Saurer, and M. Pollefeys. Live metric 3D reconstruction on mobile phones. *Proc. Int'l Conf. Computer Vision*, pages 65–72, 2013.

[26] G. Vogiatzis and C. Hernández. Video-based, real-time multi-view stereo. *Image and Vision Computing*, 29(7):434–441, 2011.

[27] A. Wendel, M. Maurer, G. Graber, T. Pock, and H. Bischof. Dense reconstruction on-the-fly. pages 1450–1457, 2012.

[28] C. Wu. VisualSFM: A Visual Structure from Motion System. `http://homes.cs.washington.edu/~ccwu/vsfm/`.