Virtual Reality (2016) 20:221–235
DOI 10.1007/s10055-016-0297-5

ORIGINAL ARTICLE

3D model reconstruction with common hand-held cameras

Maoteng Zheng[1] · Junfeng Zhu[2] · Xiaodong Xiong[2] · Shunping Zhou[1] · Yongjun Zhang[3]

Received: 13 July 2016 / Accepted: 11 September 2016 / Published online: 17 September 2016

Abstract A 3D model reconstruction workflow with hand-held cameras is developed. The exterior and interior orientation models combined with the state-of-the-art structure from motion and multi-view stereo techniques are applied to extract dense point cloud and reconstruct 3D model from digital images. An overview of the presented 3D model reconstruction methods is given. The whole procedure including tie point extraction, relative orientation, bundle block adjustment, dense point production and 3D model reconstruction is all reviewed in brief. Among them, we focus on bundle block adjustment procedure; the mathematical and technical details of bundle block adjustment are introduced and discussed. Finally, four scenes of images collected by hand-held cameras are tested in this paper. The preliminary results have shown that sub-pixel (<1 pixel) accuracy can be achieved with the proposed exterior–interior orientation models and satisfactory 3D models can be reconstructed using images collected by hand-held cameras. This work can be applied in indoor navigation, crime scene reconstruction, heritage reservation and other applications in geosciences.

Keywords 3D model reconstruction · Hand-held cameras · Bundle block adjustment · Preconditioned conjugate gradients · Multi-view stereo

✉ Maoteng Zheng
tengve@163.com

[1] National Engineering Research Center for Geographic Information System, China University of Geosciences, Wuhan, China

[2] Smart Mapping Technology Inc., Beijing, China

[3] School of Remote Sensing and Information Engineering, Wuhan University, Wuhan, China

1 Introduction

3D model reconstruction is an essential procedure in virtual reality. To create a virtual scene of the reality, the 3D model of the reality scene should be firstly reconstructed. Cameras are the most commonly used sensors for data collection in 3D model reconstruction. They are also the most familiar electronic devices around human beings. Almost every smart phone is equipped with two cameras, the front camera and rear camera. The rear camera always has a much higher resolution than the front camera, even close to the professional digital camera. Those cameras are usually hand-held by human beings. Theoretically, anyone who has a digital camera or a smart phone with high-resolution cameras can collect images to reconstruct 3D model. However, those cameras are designed only for amateur photographing. The focal length is not fixed. The lens distortion is large and unknown. Furthermore, the positions and attitudes of hand-held camera are unknown, and the imaging structure of a scene is also not regularly aligned. Those characteristics make the images collected by hand-held cameras much more difficult to be applied in 3D model reconstruction. However, those hand-held cameras are quite convenient. If the 3D model reconstruction can be implemented with these hand-held cameras, the conventional complex 3D modeling work could be easier; more people can study and participate in the 3D modeling work or even in virtual reality activities through their hand-held cameras. This is significant to the development and innovation of photogrammetry, remote sensing and virtual reality communities.

3D modeling is a relatively complex procedure. The most frequently used methods of 3D modeling are photogrammetry methods (Jesse 2015; Agisoft 2015; Acute3D 2015; Eos Software module Inc. 2015; SimActive Inc.

2015; Rothganger et al. 2006; García-Gago et al. 2014; Rau and Chen 2003; Kocaman et al. 2006; Ozaki et al. 2011; Bujnak et al. 2009; Park and Subbarao 2004; Park et al. 2008; Wang 2012; Elias and Kebisek 2010), light detection and ranging (LiDAR) methods (Ackermann 1999; Li et al. 2012; Jiang et al. 2014; Zhang et al. 2006; Yu et al. 2014; Arefi et al. 2008; Martin et al. 2010; Kato et al. 2009; Yang et al. 2013; Zhu 2014) and LiDAR combined with photogrammetry methods (Baltsavias 1999; Ma 2004; Sohn and Dowman 2007; Chen et al. 2014; Kim and Habib 2009; Susaki 2013). Except the necessary auxiliary data, the first way only uses image data, the second one uses LiDAR point cloud data, and the third way uses both images and LiDAR point cloud. Photogrammetry method utilizes cameras to collect images, tie point observations are extracted and combined with other auxiliary data to restore the relative positions, altitudes and inner parameters of each camera, and then the point cloud and 3D model of the scenes can be generated. LiDAR method uses light detection positioning technology combined with the IMU/DGPS position and orientation System (POS) to directly acquire 3D coordinates of the ground points of the scenes. It is simple and effective for extraction of the Digital Surface Model (DSM). But the instruments are very expensive. Furthermore, the edge and texture information of the scenes are missing since the point cloud is collected with a regular interval. Both photogrammetry and LiDAR methods have advantages and disadvantages; thus, the LiDAR combined with photogrammetry method is proposed to exhaustively utilize the advantages and abandon the disadvantages of these two methods. However, the registration problem between the photogrammetry and LiDAR data is still not perfectly solved. The photogrammetry method is still a feasible and widely used method.

3D model reconstruction using images includes a number of procedures. Firstly, all images are preprocessed for data standardization, and tie points are automatically identified and matched in all images. Then, the ground control points (if there is any), tie points and initial positions, attitudes, known as exterior orientation parameters (EOPs), and inner parameters, known as interior orientation parameters (IOPs), of each camera are combined in a bundle block adjustment (BBA) procedure aiming to obtain the accurate EOPs and IOPs of each camera. At last, the dense point cloud is produced with these EOPs and IOPs, and the 3D model is reconstructed with these dense point cloud and the raw images. In this paper, the common digital images are used as test data for 3D modeling experiments. The main purpose of this work is to utilize proper exterior and interior orientation models, develop a stable and efficient workflow for 3D modeling with hand-held cameras. The whole procedure, with emphasis on the mathematical and technical details of BBA with these hand-held cameras, is discussed. The experimental results of the outcome dense point cloud and the reconstructed 3D model are also presented.

2 Related works

3D model reconstruction has been comprehensively studied in the photogrammetry, remote sensing and computer vision area in recent years. As mentioned before, the most frequently used methods are photogrammetry method, LiDAR method and LiDAR combined with photogrammetry method. A lot of research works have been focused on these methods.

In the photogrammetry community, explosive growth has been made in 3D model reconstruction. 123D Catch developed by Autodesk is an open-source photogrammetry software which can extract 3D information from 2D images (Jesse 2015); Photoscan developed by Agisoft (2015) is a stand-alone software product that performs photogrammetric processing of digital images and generates 3D spatial data; Smart3DCapture developed by Acute3D (2015) can turn photos into 3D models automatically; PhotoModeler developed by Eos Systems Inc. (2015) extracts 3D measurements and models from photographs taken with an ordinary camera; and Simactive (2015) is developed for the generation of high-quality geospatial data from imagery. Most of these software packages are customized to solve certain problems, for instance, aerial triangulation, 3D model reconstruction and others. The mathematic and technical details of 3D model reconstruction using only images are also discussed. Rothganger et al. (2006) used local affine-invariant image descriptors and multi-view spatial constraints to model the 3D objects. García-Gago et al. (2014) developed a photogrammetric and computer vision-based approach for automatic 3D architectural modeling and its typological analysis. Rau and Chen (2003) proposed a robust method for reconstruction of building model from three-dimensional line segments. Most of the above works use aerial imagery. Other source images are also adopted. Kocaman et al. (2006) used high-resolution satellite images to extract 3D models of buildings. Ozaki et al. (2011) tried to develop a method for 3D modeling of dynamic remote environments using the images from two cell phone cameras and a communication network. Bujnak et al. (2009) introduced a method for 3D reconstruction from images collections with only a single known focal length. Some researchers even use only 2D images and a priori information to reconstruct 3D model. Park and Subbarao (2004) and Park et al. (2008) developed a method for automatic 3D model reconstruction based on pose estimation and integration techniques and then he reconstructed a 3D face from only a single 2D face

image based on this method. To improve the efficiency, graphic processing unit (GPU) parallel computing was introduced in. Wang (2012) built a framework for GPU 3D model reconstruction using structure from motion in his master thesis. Elias reported an overview of methods for 3D model reconstruction from 2D orthographic views. He argued that most of the design works did not lie in designing new components, but in adapting, modifying and refining existing ones (Elias and Kebisek 2010). Krasić and Pejić (2014) compared the semi-automatic and full-automatic photogrammetry method in the case study of 3D modeling for the remains of the Nis Palace.

Light detection and ranging (LiDAR) system has been widely used for 3D model reconstruction in recent years. Back in 1999, Ackermann (1999) have contributed a comprehensive analysis of the status and the expectations of airborne laser scanning system. Now in twenty-first century, a lot of works are still focused on the mathematical theory and technical detail of 3D reconstruction with LiDAR data. Some focused on 3D building model reconstruction. Li et al. (2012) developed a hierarchical contour method for automatic 3D city reconstruction with LiDAR data. Jiang et al. (2014) built a model for automatic reconstruction of multilayer building 3D contour model from airborne LiDAR point cloud. Zhang et al. (2006) introduced an automatic construction of building footprints from airborne LiDAR data. Yu et al. (2014) proposed a method to automatically reconstruct the 3D building models from segmented data based on pre-defined formal grammar and rules using laser scanning data. Arefi et al. (2008) studied the levels of detail in 3D building reconstruction from LiDAR data. Some focused on the 3D modeling of plants, vegetations and others. Martin et al. (2010) applied LiDAR point cloud in canopy surface reconstruction using Hough transformation. Kato et al. (2009) performed an implicit surface reconstruction for capturing the tree crown formation using airborne LiDAR data. Yang et al. (2013) proposed a method for 3D forest reconstruction and structural parameter retrievals using a terrestrial full-waveform LiDAR instrument. Zhu (2014) used airborne and mobile laser scanning to reconstruct 3D model of the railway environments.

Baltsavias (1999) has reported an early comparison research between the photogrammetry and laser scanning. These two methods both have advantages and disadvantages, thus combining them should be a wise choice. Ma (2004) had studied the theory and technical details of building model reconstruction from LiDAR data and aerial photographs in his doctoral dissertation. Sohn and Dowman (2007) performed the data fusion of high-resolution satellite imagery and LiDAR data for automatic 3D model of building extraction. Chen integrated LiDAR and camera data for 3D reconstruction for both indoor and outdoor environments (Chen et al. 2014). Kim and Habib (2009) studied the object-based integration of photogrammetric and LiDAR data for automatic generation of complex polyhedral building models, while Susaki (2013) proposed a knowledge-based modeling of building in dense urban areas by combining airborne LiDAR data and aerial images. Despite that a lot of research works have been done, but the registration between these two kinds of data still needs to be perfectly solved. None of the present solutions is satisfying in both stability and efficiency. Some other methods of 3D model reconstruction are also applied. For instance, Zhang et al. (2013) performed a real-time 3D model reconstruction and interaction system using Kinect for a game-based virtual laboratory.

In this work, we choose an economical and practical way, photogrammetry method. The source images are photographed by common hand-held cameras.

3 Methodology

To reconstruct 3D model from images, correspondence of images should be identified via tie point extraction and relative orientation procedure. Then, the BBA is applied to improve the accuracy of the image orientation. Finally, dense point cloud is produced using these orientation parameters and 3D models are reconstructed. Methods of processing common digital images are quite different from conventional aerial photogrammetry. A lot of the researchers have been focused on this problem. Some good methods and algorithms have been proposed, such as structure from motion (SFM) and multi-view stereo (MVS). In this paper, an efficient and effective BBA method using preconditioner conjugate gradient algorithm combined with the state-of-the-art SFM and MVS techniques is applied to reconstruct the 3D model with common hand-held cameras.

3.1 Tie point extraction and relative orientation

Common hand-held cameras are non-metric cameras. The correspondence problem of these images is difficult due to the distortions and deformations. Thus, a stable and efficient correspondence algorithm is required. Fortunately, the scale-invariant feature transform (SIFT) can provide robust feature extraction and image matching performance, invariant to many transformations such as scaling and rotating (Lowe 2004). SIFT is adopted to firstly identify feature points on the images and then match the conjugate points on the corresponding images. More information about SIFT can be found in reference (Lowe 2004). Although SIFT is invariant to many deformations, it is still prone to errors (Agarwal et al. 2011). To avoid

mismatches, the epipolar constraint is applied. Epipolar searching is an effective and efficient strategy in image matching based on the theory that the conjugate points should be on the corresponding epipolar line as shown in Fig. 1. This strategy can not only decrease the errors, but also improve the searching efficiency. To extract tie points, exhaustive matching of all the images is implemented. But this process is very time consuming especially when many images are involved (for instance, more than 1000). Then, a fast match strategy is needed. Actually, some researchers had already noticed this problem, and some solutions had also been reported. Among them, Agarwal et al. (2011) proposed a quick image matching method base on image skeleton in his research. When image number is getting bigger, his method could be a wise choice. Figures 2 and 3 show the screenshots of the tie points from two scenes.

Once the tie points are obtained, we can use them to perform the relative orientation to connect all the images and building a scalable block. The relative orientation is a classic and well-defined algorithm in the conventional photogrammetry process. Its main purpose is to acquire the relative position and attitude of the images with respect to a local coordinate system. All the images in the block can be connected using these relative positions and attitudes. In this paper, common digital images are used for 3D modeling. These images are always unordered and irregularly aligned. Some images might have no overlap with the rest ones. They should be removed from the block. In the conventional relative orientation process, the images are connected one by one. In here, a block adjustment will be performed at each time when the connected image number is increased by a certain number. This strategy is applied to avoid the error accumulation when connecting images one by one. The threshold of image numbers should be determined according to the accuracy and efficiency of the relative orientation process. Our empirical value of the certain number is 50. After relative orientation, the position and attitude of the all images in the local coordinate system can be obtained and a scalable model can be built as shown in Fig. 4.

3.2 Bundle block adjustment

BBA is to further determine the camera parameters (including EOPs and IOPs) and improve the accuracy using the tie point observations and other given information. It is a significant and essential process for 3D modeling. The accuracy of BBA can directly affect the accuracy of the reconstructed 3D model. Besides, good accuracy can largely improve the efficiency of the MVS since that the EOPs and IOPs are used to predict the positions of conjugate points on the corresponding images during the MVS process. The accuracy of camera parameters is higher, the MVS process is quicker and the final 3D model is more accurate.

3.2.1 Imaging geometry

A ground point P(X, Y, Z) is imaged by a camera with parameters (Xs, Ys, Zs, ϕ, ω, κ) known as EOPs and (f, $x0$, $y0$, $k1$, $k2$) known as IOPs. Then, an image point p(x, y) corresponding to the ground point P can be obtained in the image. The camera lens center is defined as the perspective center S. The ground point P, its corresponding image point p and the perspective center S are on the same line; the relationship can be described by formulae as Eqs. (1), (2) and (3).

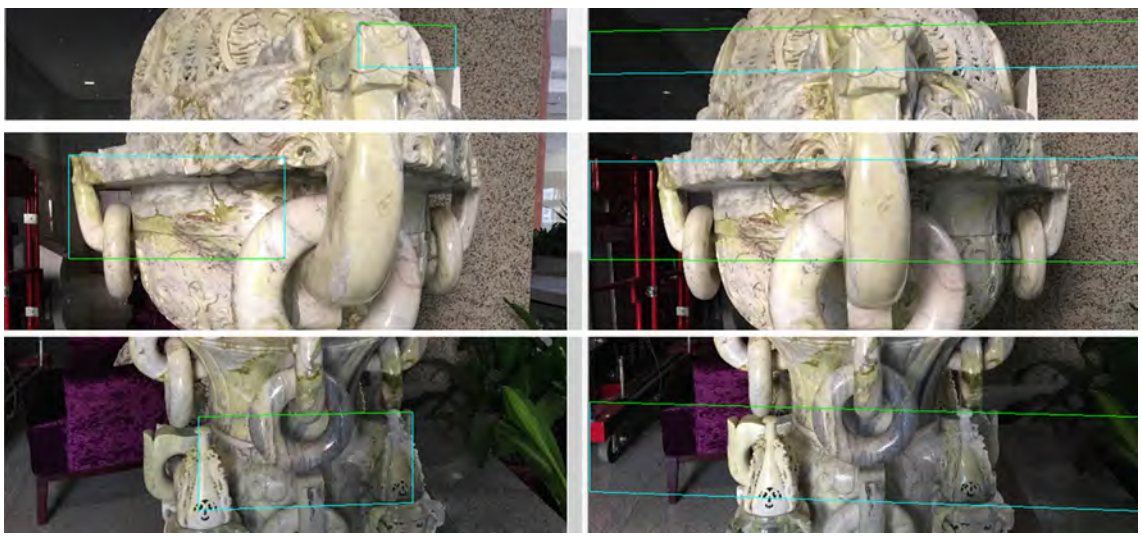

Fig. 1 Epipolar line on the *left* and *right* images, respectively

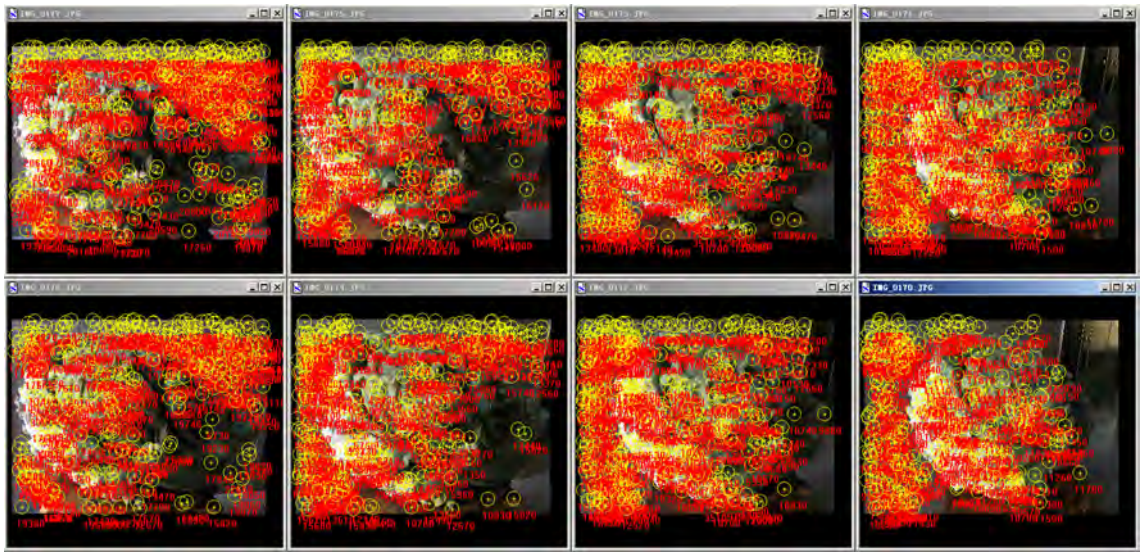

Fig. 2 Tie points shown on the test scene 1

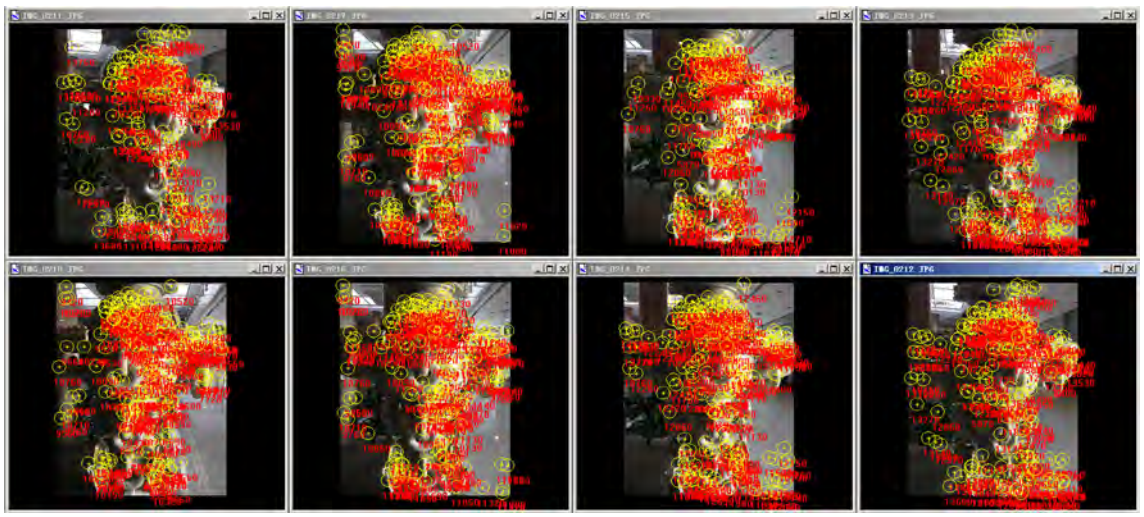

Fig. 3 Tie points shown on the test scene 2

$$\begin{bmatrix} x-\Delta x \\ y-\Delta y \\ -f \end{bmatrix} = R^T \begin{bmatrix} X-Xs \\ Y-Ys \\ Z-Zs \end{bmatrix} \tag{1}$$

$$R = R(\phi,\omega,\kappa) = \begin{bmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{bmatrix} \tag{2}$$

Combine (1) and (2) we have

$$\begin{bmatrix} x-\Delta x \\ y-\Delta y \\ -f \end{bmatrix} = \begin{bmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{bmatrix} \begin{bmatrix} X-Xs \\ Y-Ys \\ Z-Zs \end{bmatrix} \tag{3}$$

f is the focal length of the camera; it can be read out from the auxiliary data. $\Delta x, \Delta y$ in Eqs. (1) and (3) are known as corrections for image point coordinates. They can be expressed by IOPs, lens distortion parameters k_1, k_2 and principle point translation parameters x_0,y_0, as shown in Eqs. (4) and (5). This interior orientation model is applied to eliminate the distortions and other deformations in the common digital cameras.

$$\begin{cases} \Delta x = x_0 + k_1(x-x_0)r^2 + k_2(x-x_0)r^4 \\ \Delta y = y_0 + k_1(y-y_0)r^2 + k_2(y-y_0)r^4 \end{cases} \tag{4}$$

$$r = \sqrt{(x-x_0)^2+(y-y_0)^2} \tag{5}$$

3.2.2 *Solving normal equation*

To solve the EOPs and IOPs of all cameras and all the ground point coordinates (GPC) based on the collinearity

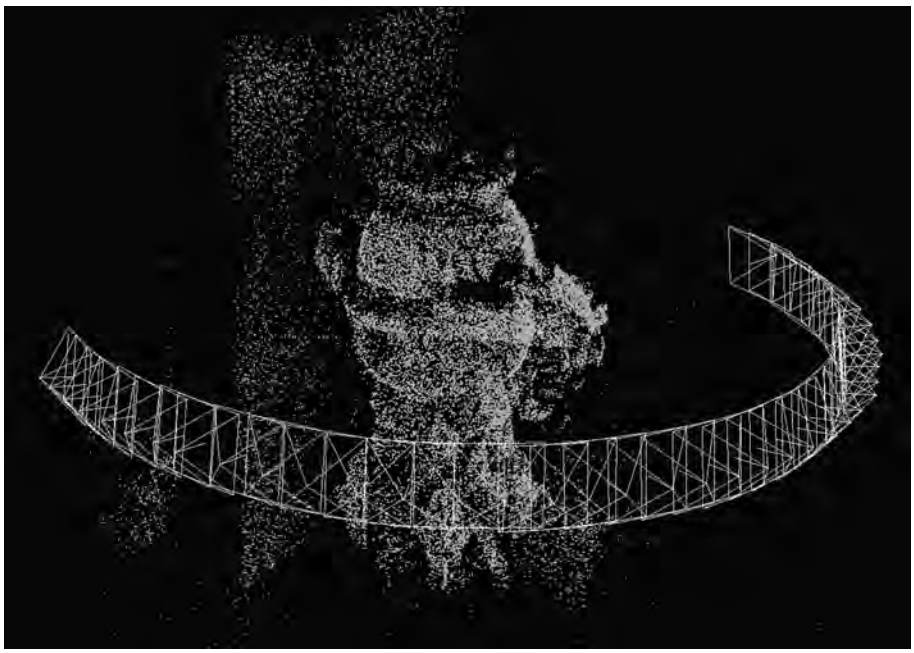

Fig. 4 A scalable model of a scene where the relative positions and attitudes of all images are demonstrated around the scene

condition, we build error equations from Eq. (3) according to the Levenberg–Marquardt (LM) model, and we have:

$$V = AX - L \tag{6}$$

where V is the residual vector, A is a matrix consist of the first-order derivatives of Eq. (3) to the unknowns (EOPs and GPC), and it is also called Jacobi matrix. X is the unknown vector. L is the discrepancy vector of the image points.

Then, we build the normal equation. Meanwhile, a damping term λD is used in case that the rank of A^TA is not full and makes Eq. (6) irresolvable. So we have Eq. (7).

$$(A^TA + \lambda D)X = A^TL \tag{7}$$

where the matrix D is usually the diagonal of matrix A^TA; λ is a damping value between (0, 1). It should be changed according to the result of each iteration.

The Jacobi matrix A can be partitioned into two parts, such as camera part and ground point part, so the matrix A can be rewritten as $A = [A_C \quad A_P]$, the same can be done to $D = [D_C \quad D_P]$ and $X = [X_C \quad X_P]$. Then, we can rewrite the normal equation as follows:

$$\begin{bmatrix} A_C^TA_C + \lambda D_C & A_C^TA_P \\ A_P^TA_C & A_P^TA_P + \lambda D_P \end{bmatrix} \begin{bmatrix} X_C \\ X_P \end{bmatrix} = \begin{bmatrix} A_C^TL \\ A_P^TL \end{bmatrix} \tag{8}$$

Let $V_C = A_C^TA_C + \lambda D_C$, $V_P = A_P^TA_P + \lambda D_P$, $W = A_C^TA_P$, $L_C = A_C^TL$, $L_P = A_P^TL$, and we have Eqs. (9) and (10).

$$\begin{bmatrix} V_C & W \\ W^T & V_P \end{bmatrix} \begin{bmatrix} X_C \\ X_P \end{bmatrix} = \begin{bmatrix} L_C \\ L_P \end{bmatrix} \tag{9}$$

$$SX_C = B \tag{10}$$

where

$$S = V_C - WV_P^{-1}W^T \tag{11}$$

$$B = L_C - WV_P^{-1}L_P \tag{12}$$

Unknown parameters X_C can be calculated by Eq. (10), and X_P can be then substituted from Eq. (9). The normal matrix size is reduced to the size of the unknown camera parameter part. This process is the so-called Schur compliment.

3.2.3 *Conjugate Gradient methods*

Conventional BBA uses LM and Schur compliment method to solve the normal equation. But when the image number is getting bigger, the normal matrix will be too large to be stored and inverted in the computer. Most researchers choose the conjugate gradient (CG) algorithm.

Conjugate gradient algorithm is firstly proposed by Hestenes and Stiefel (1952), and it is an iterative method for solving the linear symmetric positive defines system. During the iteration of the CG process, an initial vector x^0 is given as the approximate initial answer of the normal equation, and then a new vector x^1 is computed by the x^0 and other given parameters. As it repeated for certain times n, the process will eventually converged to a vector x^n which should be the final answer of the normal equation. The main advantage of CG is that it avoids matrix–matrix

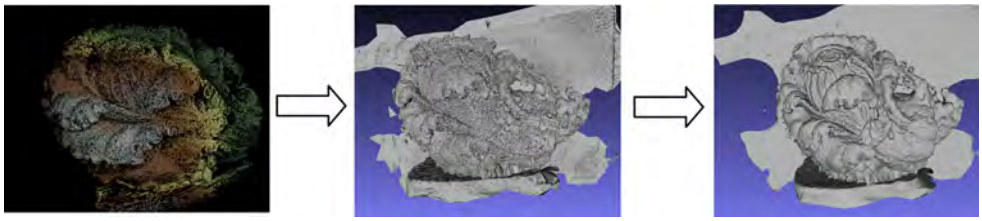

Fig. 5 Reconstruction of 3D model from dense point cloud, the left image shows point cloud, the middle shows a coarse 3D model, and the right shows a refined and accurate 3D model

Table 1 Test data information

Scenes	Number of images	Phone/camera type	Image size (pixel)	Photographed date
Human	40	Xiaomi	3120*4208	2015-04-14
Cabbage	41	iPhone 6 Plus	3264*2448	2015-01-22
Statue 1	91	iPhone 6 Plus	3264*2448	2015-01-22
Statue 2	232	Canon EOS-1Ds	5616*3744	2011-07-20

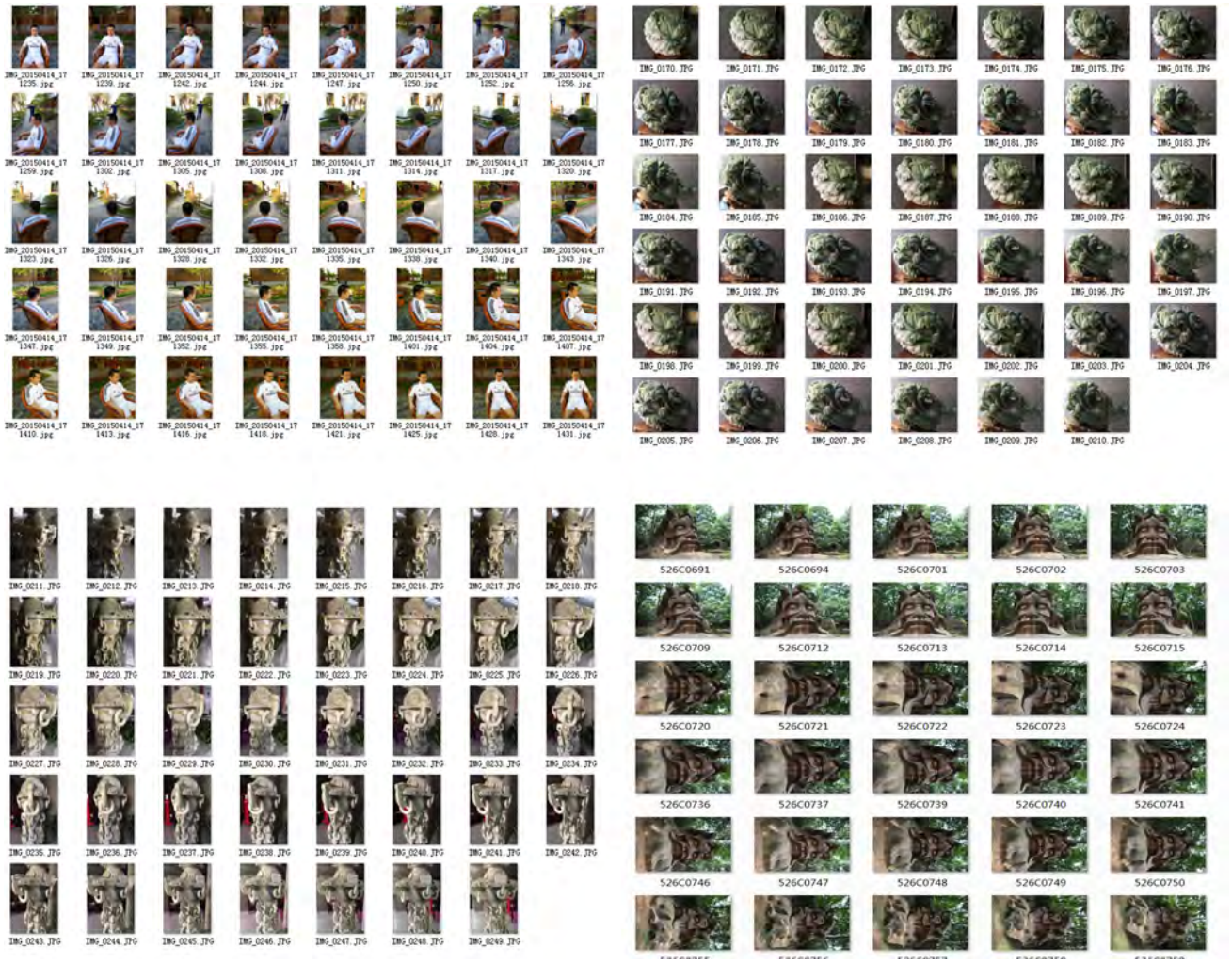

Fig. 6 Images of the four scenes

multiplications and matrix inversion which are both time-consuming computations. Only matrix–vector multiplications are needed.

The converging times are related to the condition of the normal matrix. The theoretical iteration times to convergence should be equal to the condition of the normal matrix. But it has been reported that after r (much smaller than n) times, the solution x^r will be close enough to the true answer. If one need to further improve the converging speed, a proper preconditioner should be used; this method is called preconditioned conjugate gradient (PCG). The PCG method is to apply a preconditioner M^{-1} to the normal matrix, so as to decrease the condition of the normal matrix, and thus accelerate the iteration process. After applying a preconditioner, Eq. (10) can be rewritten as follows:

$$M^{-1}Sx_c = M^{-1}l \tag{13}$$

The iteration times now should be no more than the condition of matrix $M^{-1}S$. The main task is shifted to finding a proper preconditioner which can not only decrease the condition of the normal matrix but also is easy to be inverted. The simplest and most widely used preconditioner is block Jacobi preconditioner which uses a block diagonal of the normal matrix as the preconditioner. Other preconditioners, such as symmetric successive over-relaxation (SSOR) preconditioner (Agarwal et al. 2010), QR factorization preconditioner (Byröd and Åström 2010), balanced incomplete factorization-based preconditioner (Bru et al. 2008), multiscale preconditioner (Byröd and Åström 2009) and subgraph preconditioner (Jian et al. 2011), could be more efficient but might be more complicated and less stable.

The PCG algorithm can largely decrease the memory requirement of normal equation especially for a great number of images. As reported in reference (Zheng et al. 2016), when image number is more than 5000, the memory requirement of normal equation will be more than 6.7 GB which is too large for a common computer. Despite some high-performance computer can spare this large memory space, the computation efficiency will be compromised when a large portion of RAM is occupied by normal equation. More information about this can be found in (Zheng et al. 2016).

4 3D model reconstruction

After relative orientation and BBA process, the EOPs and IOPs of the cameras are recovered. The 3D coordinates of the tie points are also obtained. But these points are not dense enough to express the 3D model. So a dense match procedure is still necessary to produce dense point cloud of the scene. Dense feature points are firstly extracted by Harris or other effective feature point extraction algorithms. Then, the well-known MVS algorithm is adopted to extract the dense 3D point cloud. In this paper, we adopt a patch-based MVS (PMVS) algorithm. The details can be found in the literature (Furukawa and Ponce 2010). The EOPs and IOPs are important orientation parameters which are used in PMVS to predict the potential positions of the conjugate points in the corresponding images. Some gross points would exist due to the low contrast and weak texture; thus, a blunder detection and elimination algorithm should be applied to remove gross points.

To build a 3D model, the dense point cloud need to be further processed. The Poisson surface reconstruction (PSR) algorithm is adopted to generate the triangulated mesh model (Furukawa and Ponce 2010). A coarse model is firstly generated which is called an initial model, and then a refined model is extracted based on this initial model and related information by abandoning the outliers according to the method proposed in the literature (Furukawa and Ponce 2010). This process is also demonstrated as in Fig. 5.

5 Experiments and analysis

5.1 Dataset

There are totally four scenes of test images which are photographed by hand-held digital cameras. The first scene is a man sitting in a chair, the second scene is a cabbage, and the third and the fourth scene are both statues. The data information is listed in Table 1. The test images are shown in Fig. 6. All these images were preprocessed before the 3D model reconstruction procedure.

In all the experiments, SIFT algorithm is implemented by the well-known OPENCV library (release 2.4.4) which

Table 2 RMSE of the reprojection error after the BBA with four scenes

Scene	Images	Points	Observations	RMSE of the reprojection error after BBA (pixels)	
				x	y
Human	40	10,042	37,244	0.417	0.522
Cabbage	41	36,633	161,396	0.554	0.557
Statue 1	91	104,075	409,812	0.514	0.509
Statue 2	232	49,913	155,894	0.637	0.443

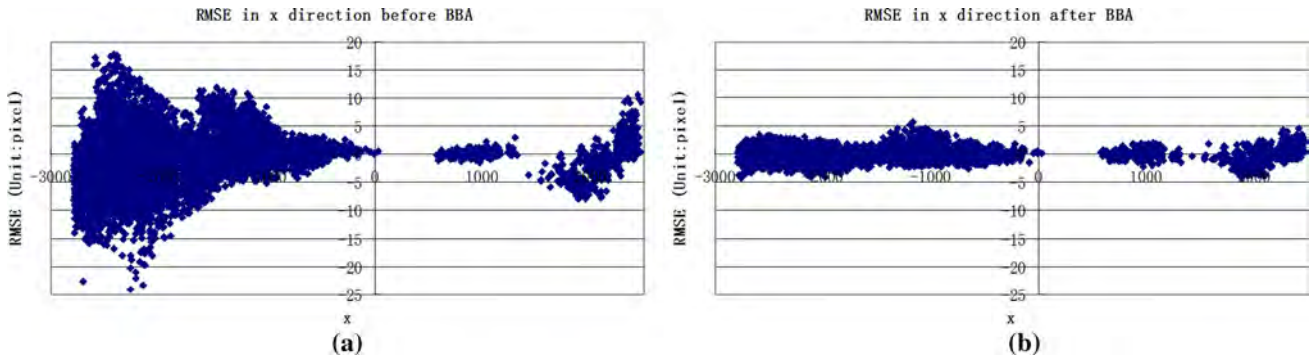

Fig. 7 The x-axis in the above figures represents the *x*-axis in image coordinate system, and the *y*-axis in **a**, **b** represents the RMSE of reprojection error in *x* direction before and after BBA, respectively

is available at http://opencv.org/. BBA module is developed by the authors according to the method mentioned in subsection *B* in section III and literature (Zheng et al. 2016). PMVS module is also developed by the authors according to the method in the literature (Furukawa and Ponce 2010). The octree depth in Poisson reconstruction process is 10. All the experiments are performed on a common laptop computer equipped with the Inter (R) Core(TM) i5-33320 M CPU 2.60 GHz, 8.00 GB RAM, and 64-bit Windows 7 operating system.

We successively performed tie point extraction, relative orientation, BBA, dense point cloud extraction and 3D model reconstruction. The test results and analysis are presented in the next two sections.

5.2 Accuracies of bundle block adjustment

Four scenes of images are tested in this paper, and the root-mean square errors (RMSE) of the image point reprojection error are shown in Table 2.

As can be seen in Table 2, after BBA, the RMSE of the image points are improved from 1 to 2 pixels to about 0.5 pixels. This is also clearly demonstrated in Fig. 7. It indicates that the interior orientation model is quite suitable for the hand-held digital camera. BBA with these images can achieve considerable sub-pixel accuracy. Thus, it is practical for 3D model reconstruction using common hand-held digital cameras.

5.3 Dense point cloud and 3D model

After BBA, the high-precision EOPs and IOPs are obtained. These parameters are then used in dense point cloud extraction. MVS uses EOPs and IOPs to predict the conjugate image points. The accuracy of EOPs and IOPs is higher, the MVS process is quicker, and the 3D model is more accurate. Four models are reconstructed with four image clusters as shown in Fig. 8.

As demonstrated in Fig. 8, all the reconstructed 3D models are basically acceptable. These models are elaborate with respect to the real object despite that some 2D features are hardly to be extracted (such as the low-contrast area in the images of statue 2). The low-contrast area as demonstrated in Fig. 9 can also be modeled well. The 3D model reconstructed by the high-resolution digital cameras (Dataset 4 in Table 2) is better than others. This is mainly because of the disparities in resolution and lens quality.

5.4 Compared to other commercial software

Two of the above scenes are also processed by Pix4D (version 1.1.38-64 bit) software (https://pix4d.com/). The main setting of parameters is shown in Table 3. The results of Pix4D and our method are shown in Figs. 10 and 11.

As shown in Figs. 10 and 11, the outcome point clouds are almost the same between Pix4D and our method, but the reconstructed models are different. There are less outliers in our method than result of Pix4D. This is mainly contributed by the refine process of our method as mentioned in subsection *C* of section III. The analysis of Pix4D result is unavailable since the specific method used in Pix4D is unknown.

To verify the accuracy, we measured some distances in 3D models reconstructed by Pix4D and our method, respectively, as shown in Figs. 12 and 13. Only relative error is valid since that the 3D models are all scalable and no control points were measured. The relative accuracy can be assessed through the comparison of our method with Pix4D. Assume that the accuracy of Pix4D has been well assessed since it is a mature commercial software. So if our method has the same accuracy to Pix4D, our method should be acceptable in accuracy phase.

It is obviously that there is a scale factor between Pix4D model and our model; different models have different scales. To compare the accuracy of our method with Pix4D, firstly the average scale factor is calculated by the following equation:

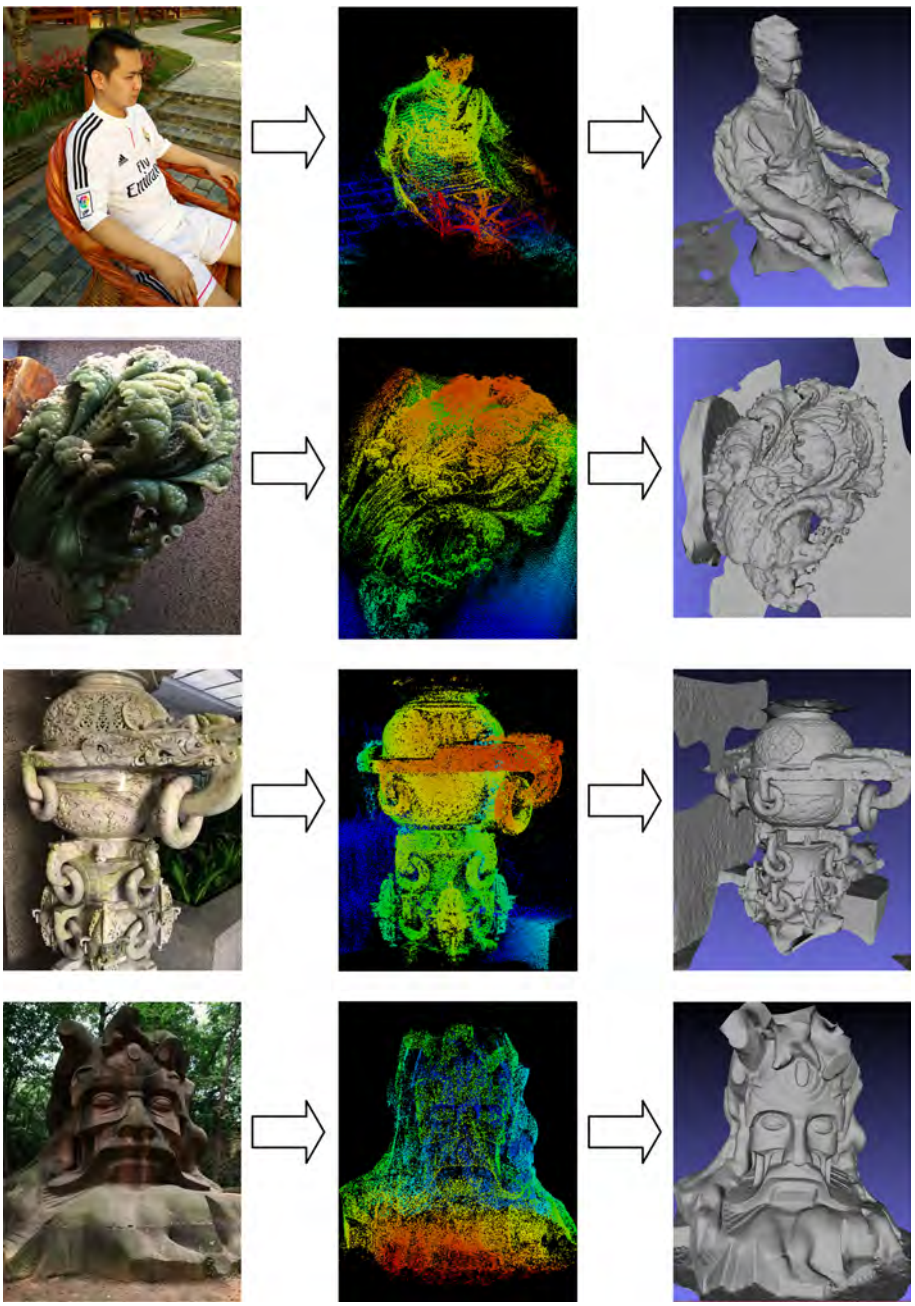

Fig. 8 3D model reconstruction of four scenes, where the *left image* is the raw image; the *middle image* is the screenshot of the 3D dense point cloud; and the *right image* is the screenshot of the reconstructed 3D model

$$\mathrm{sc} = \frac{1}{N}\sum_{i=0}^{N}\frac{L^i_{\mathrm{ours}}}{L^i_{\mathrm{pix4D}}} \tag{14}$$

where sc is the average scale factor, N is the number of measured lines, L^i_{ours} is the measured length of line i in our model, and L^i_{pix4D} is the measured length of the line i in Pix4D model.

Then, the error and relative error of our model with respect to the Pix4D model are calculated by the following equations:

$$e^i = L^i_{\mathrm{ours}} - L^i_{\mathrm{pix4D}} \cdot sc \tag{15}$$

$$r^i = e^i / L^i_{\mathrm{ours}} \tag{16}$$

where e^i is the error of line i, and r^i is the relative error of line i.

As can be seen in Table 4, the relative error of our 3D model with respect to Pix4D is about 1 % which is an acceptable accuracy. According to this relative accuracy, if the true length is 1 m, the error would be about 0.01 m. It also indicates that the 3D models reconstructed by our method are as fine as Pix4D in the accuracy phase.

Fig. 9 The reconstruction performance in the low-contrast area as shown in *red* and *blue rectangles* in the source image and 3D model, respectively (color figure online)

Table 3 Parameters setting of Pix4D in the experiment

Parameters	Options	Our setting
Point cloud density	High/optimal/low	Optimal
Minimal number of matches	2/3/4/5/6	3
Use noise filter	Yes/no	Yes
Generate triangle mesh	Yes/no	Yes

5.5 Potential applications

These 3D models reconstructed by the common hand-held camera have relatively good accuracies which have potential applications in many fields. In the indoor environment where the GPS signal is unavailable, one can walk through the indoor area while holding a camera and taking pictures. Then, the 3D model of the indoor scenes and the camera trajectory can be reconstructed and restored adopting our method. This is very useful for indoor navigation. The same can be done in a crime scene. To protect the crime scene, officers only have to take as much pictures as possible without touching any object, and high-precision 3D model of the crime scene can be reconstructed in the laboratory. If a high-resolution camera is available, the precious historical relics can be preserved by reconstructing the accurate 3D model of them using our technology. Once they are destroyed somehow, the accurate 3D model will help the engineers to rebuild or restore the relics. As reported in the literature (Krasić and Pejić 2014), the remains of Nis Palace were successfully reconstructed using photogrammetry method. Although our method is capable of dealing with large-scale data, these applications still need to be further tested with plenty of datasets.

6 Conclusion

We proposed a method for 3D modeling with common hand-held cameras. The novelty of this work is not the pure mathematical algorithm but the whole framework of the inexpensive and convenient method to reconstruct 3D model with hand-held cameras. Thus, our method can decrease the cost and might bring more people to participate into virtual reality activities which will undoubtedly promote the development of virtual reality. Besides, the PCG algorithm is introduced to solve normal equation in the bundle adjustment instead of the conventional LM model which enables our method to have potential capacity for big data (more than 5000 images in a scene). The whole procedures of 3D model reconstruction are all briefly reviewed. Totally, four scenes of images collected by hand-held cameras are tested. According to the test results and analysis, we can conclude that:

1. After BBA with the test dataset, the accuracy can reach 0.5 pixels, which indicates that the adjustment model proposed in this paper is suitable for these cameras.
2. The dense point cloud is elaborate, even in some low-contrast areas. The final 3D models are acceptable. The 3D model reconstructed by high-resolution digital camera is more elaborate than that of cameras with low resolution. After all, the common hand-held cameras have high potential for 3D model reconstruction since they are more convenient and cheaper.
3. Our experiment results are slightly better than the results of Pix4D (a commercial photogrammetry software) in some respects, while the accuracy performance are about the same.

This technology has potential applications in indoor navigation, crime scene reconstruction and heritage

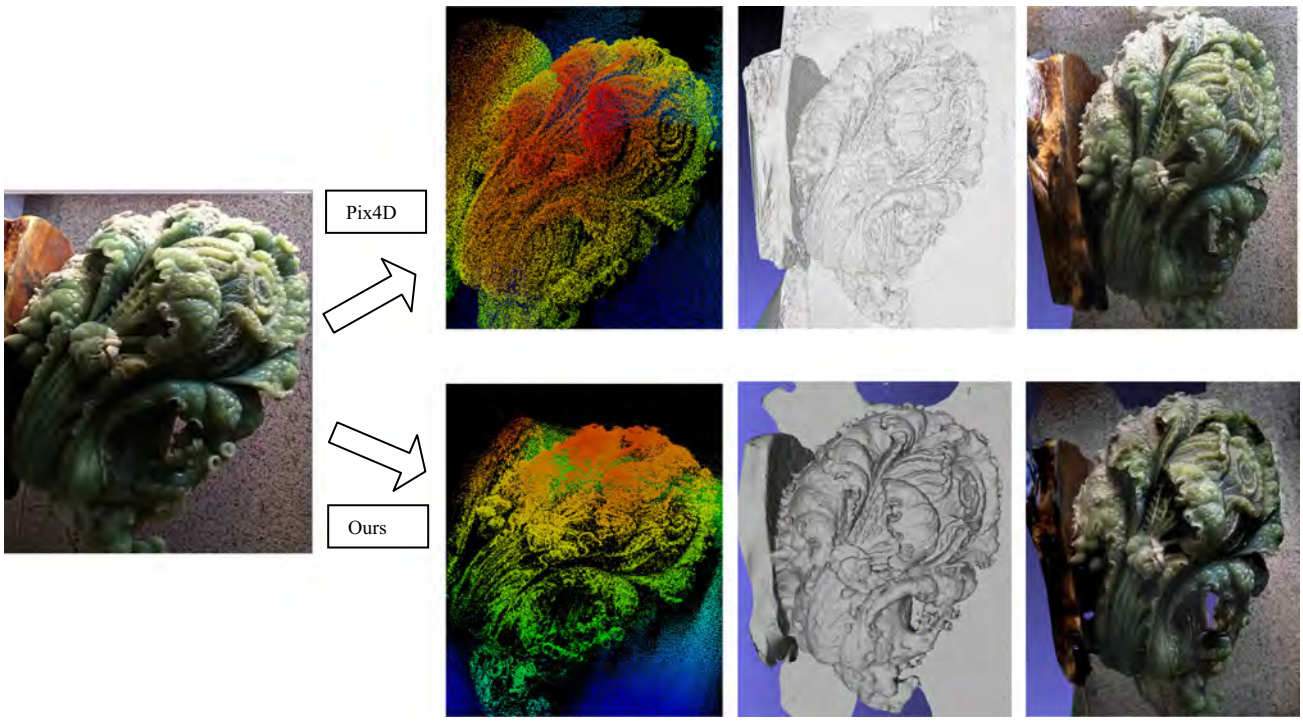

Fig. 10 3D model reconstruction of the Cabbage by Pix4D (*top*) and our method (*down*) respectively. From *left* to *right*, the *first image* is the raw image; the *second image* is the screenshot of the 3D dense point cloud; the *third image* is the screenshot of the reconstructed 3D model; and the *last image* is the screenshot of the textured 3D model

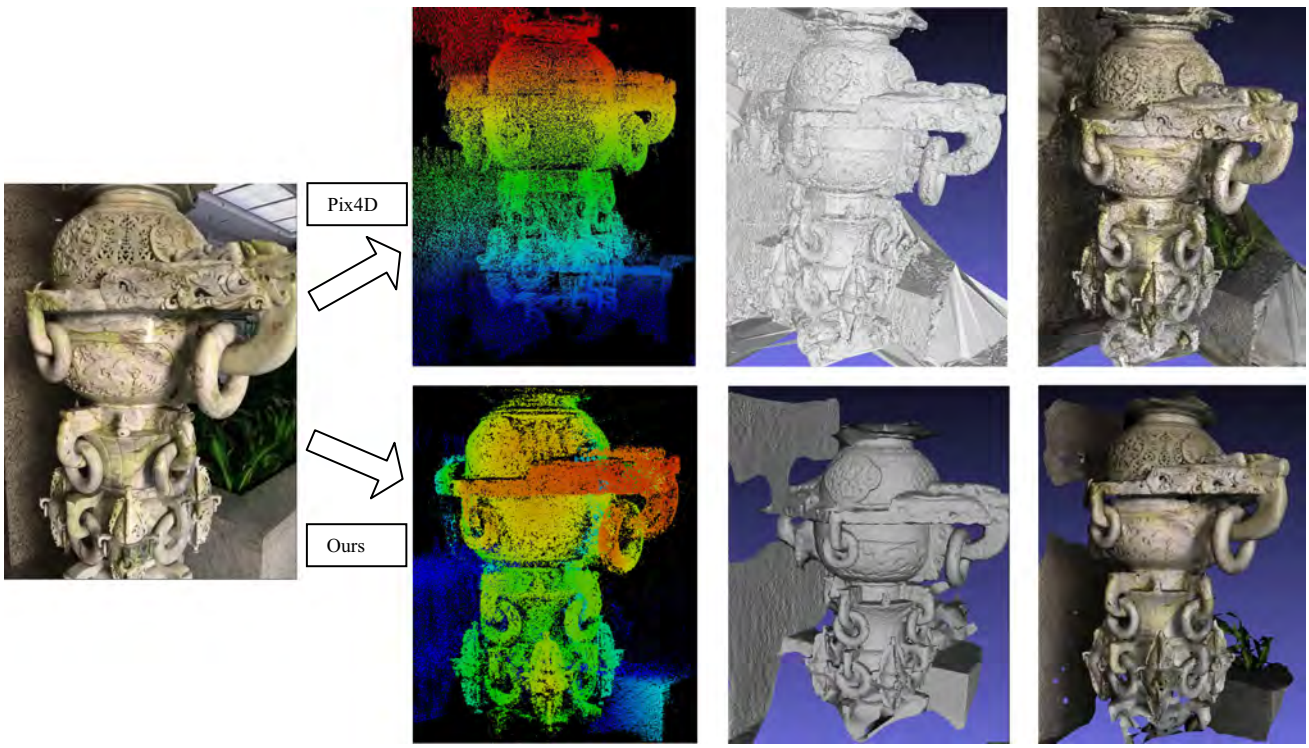

Fig. 11 3D model reconstruction of the statue 1 by Pix4D (*top*) and our method (*down*), respectively. From *left* to *right*, the *first image* is the raw image; the *second image* is the screenshot of the 3D dense point cloud; the *third image* is the screenshot of the reconstructed 3D model; and the *last image* is the screenshot of the textured 3D model

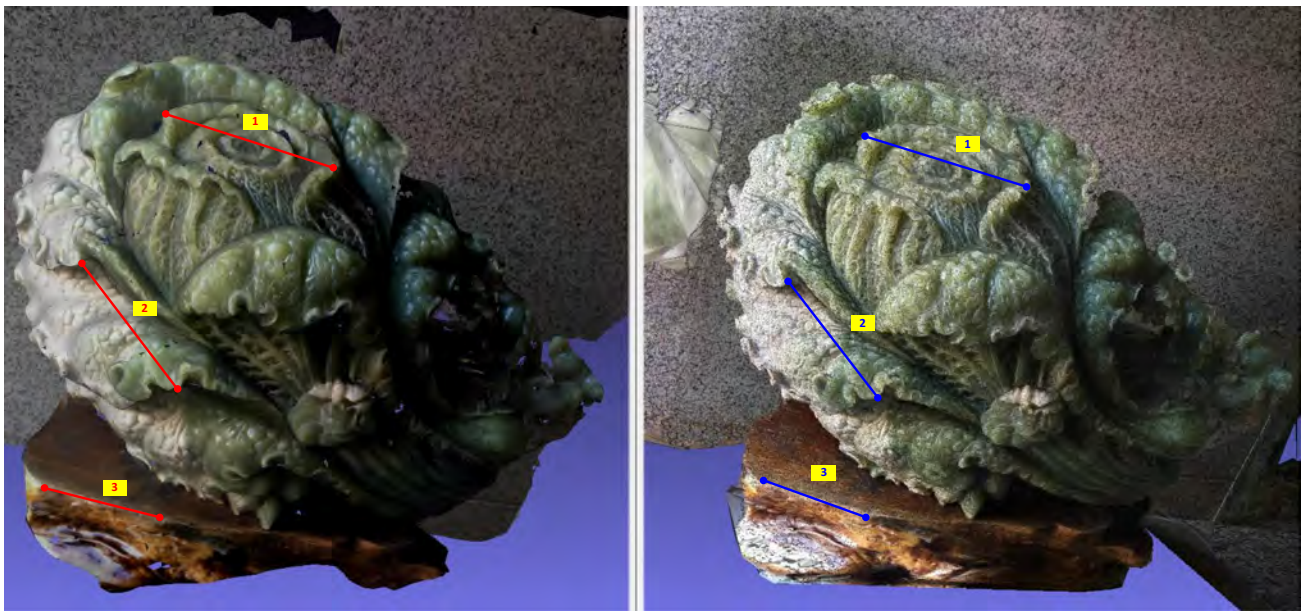

Fig. 12 Measurements in 3D model of the cabbage, *left image* shows the 3D model reconstructed by our method, *right image* shows the 3D model reconstructed by Pix4D

Fig. 13 Measurements in 3D model of the statue 1, *left image* shows the 3D model reconstructed by our method, *right image* shows the 3D model reconstructed by Pix4D

preservation for example, but the test for large-scale data is yet to be done. A lot of specific problem of large-scale data still need to be solved. This is our next research interest. The authors also have to admit that more works still need to be done for improvements on image matching strategies in occlusions and shadow areas, robustness and efficiency of BBA, gross point detection and elimination strategy in both tie points and dense point cloud.

Table 4 Comparison of accuracies of 3D models reconstructed by our method and Pix4D

Length	Cabbage			Statue		
	Line 1	Line 2	Line 3	Line 1	Line 2	Line 3
Pix4D	58.130	50.526	43.590	20.332	59.725	35.836
Our method	552.923	474.794	406.833	183.149	536.922	310.897
Average scale	9.414			8.891		
Error	5.686	−0.861	−3.521	2.375	5.897	−7.724
Relative error (%)	1.028	−0.181	−0.865	1.297	1.098	−2.485

More experiments need to be carried out to further exam and verify this work.

Acknowledgments This project is funded by the National Natural Science Foundation of China under grant 41601502 and 41571434, China Postdoctoral Science Foundation under Grant 2015M572224, the Fundamental Research Funds for the Central Universities, China University of Geosciences (Wuhan) under Grant CUG160838, and the Key Laboratory for Aerial Remote Sensing Technology of National Administration of Surveying, Mapping and Geoinformation (NASG) under Grant 2014B01.

References

Ackermann F (1999) Airborne laser scanning-present status and future expectations. ISPRS J Photogr Remote Sens 54(2):64–67

Acute3D (2015) Turn photos into 3D models automatically with Smart3DCapture. http://www.acute3d.com/smart3dcapture/. Last Accessed at 7 Sept 2015

Agarwal S, Furukawa Y, Snavely N, Simon I, Curless B, Seitz SM, Szeliski R (2011) Building Rome in a day. Commun ACM 54(10):105–112

Agarwal S, Snavely N, Seitz SM et al (2010) Bundle adjustment in the large. In: Computer vision—ECCV 2010. Springer, Berlin. pp 29–42

Agisoft (2015) Photoscan. http://www.agisoft.com/. Last Accessed at 7 Sept 2015

Arefi H, Engels J, Hahn M, Mayer H (2008) Levels of detail in 3D building reconstruction from LiDAR data. Int Arch Photogr Remote Sens Spat Inf Sci 37(B3b):485–490

Baltsavias EP (1999) A comparison between photogrammetry and laser scanning. ISPRS J Photogr Remote Sens 54(2):83–94

Bell N, Garland M (2009) Implementing sparse matrix-vector multiplication on throughput-oriented processors. In: Proceedings of the 2009 ACM/IEEE conference on supercomputing, pp 1–11

Bru R, MarÍn J, Mas J, Tůma M (2008) Balanced incomplete factorization. SIAM J Sci Comput 30(5):2302–2318

Bujnak M, Kukelova Z, Pajdla T (2009) 3D reconstruction from image collections with a single known focal length. In: IEEE 12th international conference on computer vision, 2009, pp 1803–1810

Byröd M, Åström K (2010) Conjugate gradient bundle adjustment. Lect Notes Comput Sci 6312:114–127

Byröd M, Åström K (2009) Bundle adjustment using conjugate gradients with multiscale preconditioning. In: British machine vision conference

Chen TI, Zhang YX, Chen LY et al (2014) Integration of LiDAR and camera data for 3D reconstruction. In: IEEE international conference on consumer electronics-Taiwan, pp 93–94

Elias M, Kebisek M (2010) An overview of method for 3D model reconstruction from 2D orthoraphic views. In: Proceedings of the 3rd international workshop "innovation in information technologies—theory and practice", Sept 6th–10th, Dresden, Germany, pp 1–5

Eos Software module Inc. (2015) Accurate and affordable 3D modeling—measuring—scanning. http://www.photomodeler.com/index.html. Last Accessed at 7 Sept 2015

Furukawa Y, Ponce J (2010) Accurate, dense, and robust multiview stereopsis. IEEE Trans Pattern Anal Mach Intell 32(8):1362–1376

García-Gago J, González-Aguilera D, Gómez-Lahoz J et al (2014) A photogrammetric and computer vision-based approach for automated 3D architectural modeling and its typological analysis. Remote Sens 6(6):5671–5691

Hestenes MR, Stiefel E (1952) Methods of conjugate gradients for solving linear system. J Res Natl Bureau Stand 49(6):409–436

Jesse (2015) Open source photogrammetry: ditching 123D catch. http://wedidstuff.heavyimage.com/index.php/2013/07/12/open-source-photogrammetry-workflow/. Last Accessed at 7 Sept 2015

Jian YD, Balcan DC, Dellaert F (2011) Generalized subgraph preconditioners for large-scale bundle adjustment. IEEE Int Conf Comput Vis 6669:295–302

Jiang T, Luo S, Zhang R (2014) Automatic reconstruction of multilayer building 3D contour model from airborne LiDAR point clouds. In: ISPRS technical commission I symposium, sustaining land imaging: UAVs to satellites. 17–20 Nov 2014, Denver, Colorado, USA, MTSTC1-9

Kato A, Moskal LM, Schiess P et al (2009) Capturing tree crown formation through implicit surface reconstruction using airborne LiDAR data. Remote Sens Environ 113(6):1148–1162

Kim C, Habib A (2009) Object-based integration of photogrammetric and LiDAR data for automated generation of complex polyhedral building models. Sensors 9(7):5679–5701

Kocaman S, Zhang L, Gruen A, Poli D (2006) 3D city modeling from high-resolution satellite images. In: ISPRS workshop on topographic mapping from space

Krasić S, Pejić P (2014) Comparative analysis of terrestrial semi-automatic and automatic photogrammetry in 3D modeling process. Nexus Netw J 16:273–283

Li HY, Yang C, Wang Z et al (2012) A hierarchical contour method for automatic 3D city reconstruction from LiDAR data. In: IEEE international geoscience and remote sensing symposium, pp 463–466

Lowe D (2004) Distinctive image features from scale-invariant keypoints. Int J Comput Vis 60(2):91–110

Ma RJ (2004) Building model reconstruction from LiDAR data and aerial photographs. Doctoral Dissertation for the Degree Doctor of Philosophy in the Graduate School of the Ohio State University

Martin VL, Nicholas CC, Michael AW (2010) Canopy surface reconstruction from a LiDAR point cloud using hough transform. Remote Sens Lett 1(3):125–132

Ozaki M, Tan JK, Kim H et al (2011) 3-D modeling of dynamic remote environment employing the images from cell-phone cameras and a communication network. In: SICE annual conference, 2011 proceedings of IEEE, pp 48–51

Park SY, Subbarao M (2004) Automatic 3D model reconstruction based on novel pose estimation and integration techniques. Image Vis Comput 22(8):623–635

Park SW, Heo J, Savvides M (2008) 3D face reconstruction from a single 2D face image. In: IEEE computer society conference on computer vision and pattern recognition workshops, pp 1–8

Rau JY, Chen LC (2003) Robust reconstruction of building models from three-dimensional line segments. Photogr Eng Remote Sens 69(2):181–188

Rothganger F, Lazebnik S, Schmid C et al (2006) 3D object modeling and recognition using local affine-invariant image descriptors and multi-view spatial constraints. Int J Comput Vis 66(3):231–259

SimActive Inc. (2015) Simactive announces correlator3D^{TM} version 6.2. http://www.simactive.com/en. Last Accessed at 7 Sept 2015

Sohn G, Dowman I (2007) Data fusion of high-resolution satellite imagery and LiDAR data for automatic building extraction. Int J Photogr Remote Sens 62(1):43–63

Susaki J (2013) Knowledge-based modeling of buildings in dense urban areas by combining airborne LiDAR Data and aerial images. Remote Sens 5(11):5944–5968

Wang Y (2012) A framework for GPU 3D model reconstruction using structure-from-motion. Dissertations & Theses submitted to the Faculty of the Graduate School of the University of Maryland

Wu C (2011) Multicore bundle adjustment. In: IEEE conference on computer vision and pattern recognition, pp 3057–3064

Yang XY, Strahler AH, Schaaf CB et al (2013) Three-dimensional forest reconstruction and structural parameter retrievals using a terrestrial full-waveform LiDAR instrument. Remote Sens Environ 135:36–51

Yu Q, Helmholz P, Belton D et al (2014) Grammar-based automatic 3D model reconstruction from terrestrial laser scanning data. In: The international archives of the photogrammetry, remote sensing and spatial information sciences, XL-4, pp 335–340

Zhang KQ, Yan JH, Chen SC (2006) Automatic construction of building footprints from airborne LIDAR data. IEEE Trans Geosci Remote Sens 44(9):2523–2533

Zhang Z, Zhang M, Chang Y et al (2013) Real-time 3D model reconstruction and interaction using kinect for a game-based virtual laboratory. In: Proceedings of the ASME 2013 international mechanical engineering congress & exposition, pp 1–8

Zheng MT, Zhang YJ, Zhou SP, Zhu JF, Xiong XD (2016) Bundle block adjustment of large-scale remote sensing data with block-based sparse matrix compression combined with preconditioned conjugate gradient. Comput Geosci 92(1):70–78

Zhu L (2014) The use of airborne and mobile laser scanning for modeling railway environments in 3D. Remote Sens 6(4): 3075–3100