Structure from motion obtained from low quality images in indoor environment

Structure from motion obtained from low quality images in indoor environment

Bojan Dikovski, Petre Lameski, Eftim Zdravevski, Andrea Kulakov
Faculty of Computer Science
and Engineering
University Ss. Cyril and Methodius
Skopje, R. of Macedonia
Email: b.dikovski@yahoo.com, {petre.lameski, eftim.zdravevski, andrea.kulakov}@finki.ukim.mk

***Abstract*—Structure from motion is the process of extracting 3D structure from images taken through the motion of the camera. The result is dependent on the quality and resolution of the images that are being taken, so it is beneficial to use as high quality images as possible. Sometimes it is not possible to obtain high level of detail in photos because of the environmental, economic or other restrictions. In this paper we analyze structure from motion when using a low resolution camera in indoor environment. The obtained results are compared with the same process when using images of higher resolution and with the 3D structure created from points taken with the Microsoft Kinect sensor.**

Keywords - structure from motion, sparse reconstruction, low resolution images, Kinect.

I. Introduction

The process of extracting geometric structures from images taken through a camera motion has an extensive research history and already a few commercial systems are available [1][2]. The most notable work publsihed on this topic is the book *"Multiple View Geometry in Computer Vision"* [3], in chapters 9 through 12. Obtaining three dimensional structure from motion (SFM) is a similar problem with finding the structure from stereo (or multiview) vision. The difference is that in stereo vision we know what is the motion between the cameras, while in SFM this is not known. Calibrated stereo rigs in theory provide better and more accurate reconstruction, but SFM has an advantage in more simplistic recording procedure which is one of the main motivations to use it in this work.

Python Photogrammetry Toolbox and GUI (PPT) [4] is an open source software package that incorporates different tools needed to perform a SFM reconstruction. It includes the SIFT algorithm [5] to detect local features in the images which are then used by Bundler [6]. Bundler is a software that reconstructs the scene incrementally using a modified version of Sparse Bundle Adjustment [7]. The output of this is a sparse point cloud, but a cloud consisting of denser points can also be made using another software package called Patch-based Multi-view Stereo (PMVS2) [8]. PPT includes the PMVS2 package, as well as the Clustering Views for Multi-view Stereo (CMVS) software which can be used for preprocessing before PMVS2. Our intent is to explore the feasibility and performance of a SFM system with low to medium quality images of an indoor environment. Using free and open source software means that anyone could easily perform the reconstruction made in our experiment producing a 3D model of any normal household item while using images made with the camera of a low-end smartphone.

To measure the results from our experiments we chose to use the Microsoft Kinect [9] sensor for providing the ground truth data. It is a low cost device that comes with a RGB camera, depth and audio sensors. The accuracy of the Kinect depth data is comparable to a laser scanning data and does not contain large systematic errors which was shown through theoretical and experimental accuracy analysis in [10]. In this work it was concluded that for mapping applications the data should be acquired in the range of 1-3 meters distance from the sensor. This range is optimal for our experiment and the setup that we used. In [11] quantitative comparison of the Kinect accuracy with stereo reconstruction from SLR cameras and a 3D-TOF camera is done. The authors have concluded that the Kinect sensor performed close to, and in some cases overperformed, the other types of reconstruction.

The rest of this paper is organised in the following way: in section 2. we talk about related work in this research area, in section 3. we explain our experiment in great detail, in section 4. we show the results of the experiment, and finally in section 5. we give a conclusion.

II. Related Work

Structure from motion in the past has been applied mostly for estimation and remaking of three dimensional structures from images made in outdoor enviroments, usually at large distances from the target structure. [12] presents an approach for modeling and rendering existing architectural scenes from sparse sets of still photographs. This approach combines an interactive photogrametric modeling method and a model-based stereo algorithm which can create realistic views of architectural scenes even further away from the original photographs. In [13] a method with *O(n)* complexity has been proposed for organizing an unordered image set into clusters of related images from the same scene. The process of clustering is based on finding matches between the image features, similar to what is done in the process of structure from motion estimation in this paper. [14] introduces an automated large-scale image registration system that was used to create a large image dataset of the MIT campus. This data has been used in research for image-based rendering and 3D reconstruction. In [15] a framework based on a Bayesian model is used for automatic acquisition of three dimensional architectural models from short image sequences. Here an object recognition approach learns of the objects identity which can then be used to

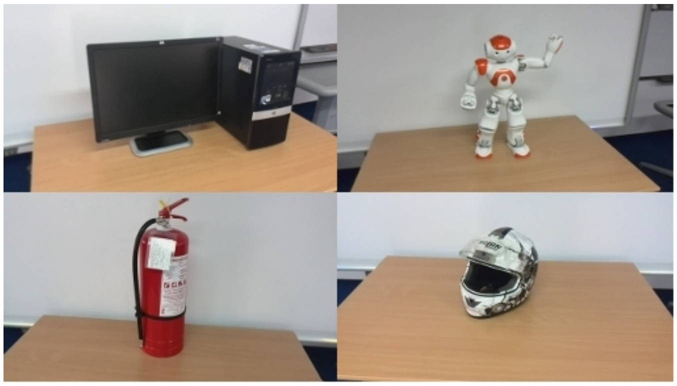

Fig. 1: *Different items used for reconstruction.*

extract information about its structure and label different parts of it. Another fully automated approach is presented in [16] where SIFT features are extracted from an unordered collection of images and used to find matching images of the same scene. Connected components of image matches are calculated and then bundle adjustment is done to solve the camera and structure parameters. Once this is done a 3D model can be generated. This kind of structure from motion reconstruction was applied in [17] and [18]. Their work produced a system for interactively browsing and exploring large unstructured collections of photographs of a scene. This system was made using Bundler, the software which we include in our work in this paper.

III. Experiment Overview

A. Data Collection

We tried to reconstruct the structure of four different items: a monitor and a chassis of a personal computer, a Nao humanoid robot, a fire extinguisher and a motorcycle helmet. These items were placed on a desk in front of a white wall in a brightly lighted room (Figure 1.). They were recorded with the person recording walking along arcs of approximately 180 degrees in front of the items. We did more than one pass, each at a different distance of the items that was in the range between 0.5 and 2 meters. Both the RGB and depth streams of the Kinect were recorded at 640 x 480 pixels resolution at a framerate of around 30 fps. Besides using the Kinect sensor, we captured video recordings using the camera of a Huawei Ascend P6 smartphone [19]. The video of the Ascend P6 was done in a resolution of 1280 x 720 pixels at a framerate of around 30 fps. The quality of the camera of Ascend P6 is moderate compared to the best smartphones available when this work was made, but still it provided noticable higher quality than the RGB Kinect stream and gave us a good measure of the available average smartphone camera. From the video sequences of both the Kinect and Ascend P6 we extracted frame images. We extracted one image per 3-5 frames automatically, and then from the those initial image sets manually we selected the actual images on which reconstruction was to be performed. The video editing, and all following reconstruction work, was done on a computer with Intel Core i3-3217U CPU, 8GB of RAM and Nvidia GeForce GT 635M graphic card.

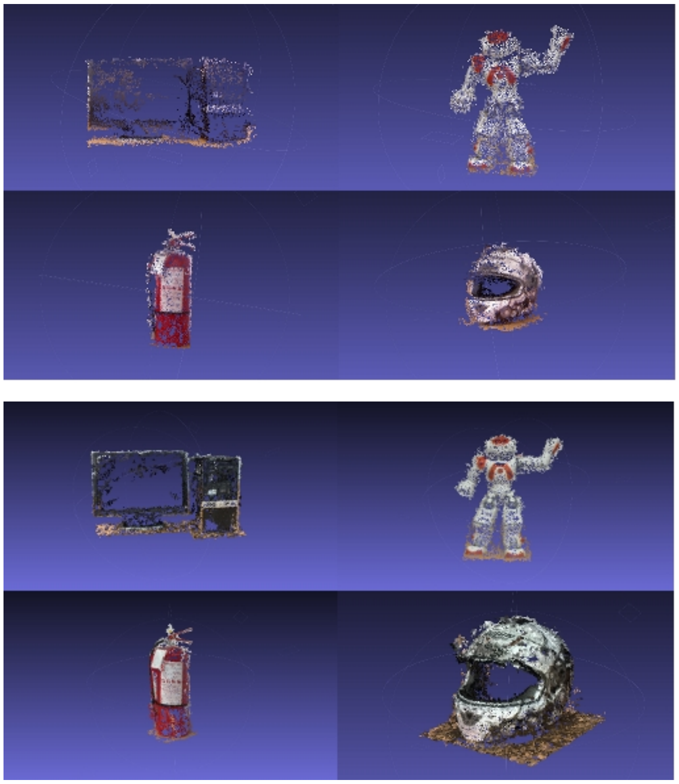

Fig. 2: *Reconstruction results from the Structure from Motion process after applied Patch-based Multi-view Stereo processing. (The top images are generated from Kinect Sensor photos while the bottom ones are generated from photos of the camera of Ascend P6.)*

B. Structure from Motion Reconstruction

For each test item we used a data set of exactly 40 images (with both test cameras). Using a lower number of images gave worse results as expected, and increasing the number of images in most cases gave better 3D reconstructed model, but also incresed the running time of the algorithm. This number was chosen as it gave satisfying results without needed a very long running time. To perform structure from motion reconstruction with Bundler, two camera parameters need to be provided as input - the camera CCD width and focal length in millimeters. The parameters that we obtained and used were 5.954 mm CCD width and 4.884 mm focal length for the Kinect, and 4.800 mm CCD width and 3.979 mm focal length for the Ascend P6. Bundler provides a choice between two image feature extractors - 'siftvlfeat' and 'siftlowe'. In this work while experimenting with different parameters and data sets it was found that 'siftvlfeat' performs better, so all our reported results are done with this feature extractor. The last thing to set before doing reconstruction is the scaling parameter. A value of 1 was used for scaling which means that the pictures were used in their original resolution. After the reconstruction was done the camera parameters were used to get a denser

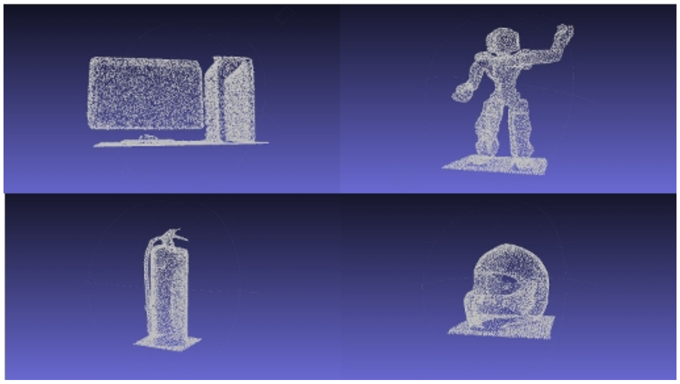

Fig. 3: *Reconstruction results from the Kinect Fusion software.*

TABLE I: Number of points in the different point clouds

Point cloud source	Desktop computer	Nao robot	Fire extin-guisher	Motorcycle helmet
SFM with Kinect Sensor	4.828	4.665	3.287	3.932
SFM with Ascend P6	23.951	10.439	11.880	32.382
Kinect Fusion	17.206	9.687	6.505	7.718

point cloud using PMVS2. Because we used only 40 images, which is a relatively low number, dense reconstruction was done using only PMVS2 without CMVS. The resulting point clouds included noise points which were easily removed using MeshLab [20]. It is a free and open-source software made for processing in the 3D scanning pipeline. The final point clouds from images of both used devices are shown in Figure 2., and the number of points of which the clouds consist are listed in Table 1.

C. Reconstruction with Kinect Fusion

To obtain the ground truth data we used Kinect Fusion [21]. This is a real-time mapping software that can be used in indoor enviroments in various lighting conditions. When using Kinect Fusion to create the 3D mesh the default parameters were used except for the depth threshold which was lowered to 3 meters. The point clouds that we got were very dense, all of the four scenes were on the scale of about one million points. After selecting out the items from the whole scene in MeshLab, we used Poisson disc-sampling to lower the number of points in the cloud. In doing this we did not lose a lot on the detail level of the model, while lowering the complexity of the following tasks to be performed. The results were very accurate and are shown in Figure 3. The number of points in each of the clouds is listed in Table 1.

IV. Results

The first result that we got from our experiment is that doubling the resolution of the photographs on which structure from motion was performed and increasing their quality resulted in greatly increased number of points in the generated point clouds. The denser clouds had almost 5 times more points for the desktop computer reconstruction, double number of points for the Nao robot model, 3.5 times more points for the fire extinguisher and over 8 times more points for the motorcycle helmet. To give further conclusion of our experiment we needed to compare the structure of the point clouds derived from the SFM process with the ground truth data from the Kinect sensor. For this we used the 3D point cloud and mesh processing software CloudCompare [22]. The first step was to bring the two points clouds that we were comparing to the same size scale. Two very distinct points found in both of the clouds that are as much further away from each other were selected and the distance between them was measured. Using this distance we could find the ratio for scaling the point clouds. The cloud that is being measured was scaled up or down to the same dimension of the ground truth cloud. The next step was to do registration of the clouds so that we can minimize the distance between them. To achieve this the Iterative Closest Points (ICP) algorithm was used [23]. After this was done the tool for computing between-cloud distance in CloudCompare was used. The results that were obtained are shown in Table 2. In Figure 4. the reconstructed models from the photographs of the Ascend P6 are shown with their points coloured according to their distance from the ground truth data. These results show that in both cases the reconstructions that were done have the same shape and are good representation of the real life items.

V. Conclusion

In this work we have evaluated the performance of a semi-automated structure from motion process done in an indoor environment. 3D models of different items were reconstructed using images from different camera sources. The results that were obtained prove our initial thesis that even with a low quality camera we can create a 3D point cloud that represents the real life model well. The framework that we used is based on free and open source software so this experiment can be performed easily without a need for some kind of monetary investment, which in time when systems like 3D printing become readily avaialable can be quite compelling. The tests that we did also showed that increasing the quality of the camera considerably increases the quality of the reconstruction. Another way to create a better model is to use a bigger number of images. However, in both cases the execution time of the process will increase. In the future working to automate and improve some of the steps that we did could provide even greater benefit to anyone who would like to create a 3D model of a everyday scene.

Acknowledgements

This work was partially financed by the Faculty of Computer Science and Engineering at the Ss. Cyril and Methodius University in Skopje, Skopje, R. Macedonia.

TABLE II: Measured distance metrics between the reconstructed point clouds and the ground truth data

	Kinect Desktop	Kinect Robot	Kinect Fire E.	Kinect Helmet	Ascend P6 Desktop	Ascend P6 Robot	Ascend P6 Fire E.	Ascend P6 Helmet
Min dist.	0	0	0	0	0	0	0	0
Max dist.	0.0618389	0.0802873	0.0823419	0.0487729	0.12322	0.0612998	0.0963241	0.0786461
Mean dist.	0.00978836	0.014143	0.0292939	0.00630127	0.0109317	0.00649334	0.0100716	0.0122751
Sigma	0.00757192	0.0124609	0.0202079	0.00490016	0.0115339	0.0058852	0.00979317	0.0107587
Max relative error	3.77496 + 0.66256/d % (d > 0.0066256)	3.77496 + 0.481724/d % (d > 0.00481724)	3.77496 + 0.525587/d % (d > 0.00525587)	3.77496 + 0.375176/d % (d > 0.00375176)	3.77496 + 0.684554/d % (d > 0.00684554)	3.77496 + 0.510832/d % (d > 0.00510832)	3.77496 + 0.535134/d % (d > 0.00535134)	3.77496 + 0.386784/d % (d > 0.00386784)

REFERENCES

[1] Autodesk 123D - *http://www.123dapp.com/catch*

[2] 2d3 Sensing - *http://www.2d3.com/*

[3] R. Hartley, A. Zisserman. *Multiple view geometry in computer vision*, Cambridge university press, 2003.

[4] Python Photogrammetry Toolbox and GUI - *http://www.arc-team.homelinux.com/arcteam/ppt.php*

[5] D.G. Lowe, *Distinctive image features from scale-invariant keypoints*, International journal of computer vision, (volume:60 issue:2 p.91–110) 2004.

[6] Bundler - *https://www.cs.cornell.edu/ snavely/bundler/*

[7] M.I.A. Lourakis, A.A. Argyros, *SBA: A Software Package for Generic Sparse Bundle Adjustment*, ACM Trans. Math. Software, (volume:36, issue:1, p.1–30) 2009.

[8] Y. Furukawa, J. Ponce. *Accurate, dense, and robust multi-view stereopsis*, IEEE Trans. on Pattern Analysis and Machine Intelligence, (volume:32, issue:8, p.1362–1376) 2010.

[9] Microsoft Kinect (http://www.microsoft.com/en-us/kinectforwindows/)

[10] K. Khoshelham, *Accuracy analysis of kinect depth data*, ISPRS workshop laser scanning (volume:38, issue:5) 2011.

[11] J. Smisek, M. Jancosek, T. Pajdla. *3D with Kinect*, Consumer Depth Cameras for Computer Vision. Springer London, (p.3–25) 2013

[12] P.E. Debevec, C.J. Taylor, and J. Malik. *Modeling and rendering architecture from photographs: A hybrid geometry-and image-based approach.*, Proceedings of the 23rd annual conference on Computer graphics and interactive techniques. ACM, 1996.

[13] F. Schaffalitzky, A. Zisserman. *Multi-view matching for unordered image sets, or How do I organize my holiday snaps?*, Computer VisionECCV 2002. Springer Berlin Heidelberg, (p.414–431) 2002.

[14] S. Teller, M. Antone, Z. Bodnar, M. Bosse, S. Coorg, M. Jethwa, N. Master. *Calibrated, registered images of an extended urban area.*, International journal of computer vision, (volume:53, issue:1, p.93–107) 2003.

[15] A.R. Dick, T.HS Philip, R. Cipolla. *Modelling and interpretation of architecture from several images.*, International Journal of Computer Vision, (volume:60, issue:2 p.111–134) 2004.

[16] M. Brown, D.G. Lowe. *Unsupervised 3D object recognition and reconstruction in unordered datasets*, Fifth International Conference on 3-D Digital Imaging and Modeling, 3DIM 2005, IEEE, (p.56–63) 2005.

[17] N. Snavely, S.M. Seitz, R. Szeliski. *Photo tourism: exploring photo collections in 3D*, ACM transactions on graphics (TOG), (volume:25, issue:3, p.835–846) 2006.

[18] N. Snavely, S.M. Seitz, R. Szeliski. *Modeling the world from internet photo collections*, International Journal of Computer Vision, (volume:80, issue:2, p.189–210) 2008.

[19] Huawei Ascend P6 - *http://consumer.huawei.com/en/mobile-phones/tech-specs/p6-u06-en.htm#anchor*

[20] P. Cignoni, M. Corsini, G. Ranzuglia. *Meshlab: an open-source 3d mesh processing system*, Ercim news, (volume:73, p.45-46) 2008.

[21] R.A. Newcombe, A.J. Davison, S. Izadi, P. Kohli, O. Hilliges, J. Shotton, A. Fitzgibbon. *KinectFusion: Real-time dense surface mapping and tracking*, In Mixed and augmented reality (ISMAR), 2011 10th IEEE international symposium on (p.127-136) 2011.

[22] CloudCompare - *http://www.danielgm.net/cc/*

[23] A.W. Fitzgibbon, *Robust registration of 2D and 3D point sets*, Image and Vision Computing, (volume:21, issue:13, p.1145-1153) 2003.

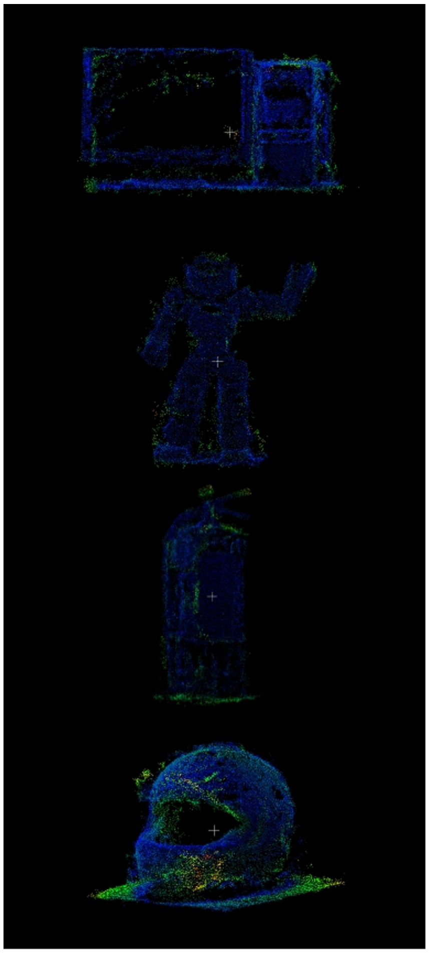

Fig. 4: *The reconstructed models from the Ascend P6 photographs. The color of the points describes their distance from the ground truth data. (blue > green > yellow > red is in the direction of min > max).*