Evaluation of Multi-view 3D Reconstruction Software

Julius Schöning(✉) and Gunther Heidemann

Institute of Cognitive Science, University of Osnabrück, Osnabrück, Germany
{juschoening,gheidema}@uos.de

Abstract. A number of software solutions for reconstructing 3D models from multi-view image sets have been released in recent years. Based on an unordered collection of photographs, most of these solutions extract 3D models using structure-from-motion (SFM) algorithms. In this work, we compare the resulting 3D models qualitatively and quantitatively. To achieve these objectives, we have developed different methods of comparison for all software solutions. We discuss the perfomance and existing drawbacks. Particular attention is paid to the ability to create printable 3D models or 3D models usable for other applications.

Keywords: Multi-view 3D reconstruction · Benchmark · Structure from Motion (SFM) · Software comparison · Photogrammetry

1 Introduction

In the last decade, a huge number of reconstruction software solutions for multi-view images has been released. This trend was boosted by the development of 3D printers. While current software solutions like *Autodesk 123D Catch* [3] and *Agisoft PhotoScan* [2] promised, e.g., to create printable models out of image collections. However, the number of printed replicas is still fairly low. In this paper we provide an overview of existing 3D reconstruction software solutions and benchmark them against each other.

The main aim of this research is to rank the four most common 3D reconstruction software solutions in a benchmark. We propose a method to objectively quantify the quality of the 3D models produced by these software solutions. Based on two different multi-view image data sets [25,29], we evaluate the solutions with respect to practical applicability in one real scenario and one planned shooting scenario. We provide objective evaluation indicators regarding both the qualitative and quantitative results of each image-based 3D reconstruction software.

The paper is structured as follows: First we give a brief overview of related work with respect to existing benchmarks and evaluations. Then, we describe the used multi-view software solutions and give the reasons for our selection of data sets. After describing the model generation and our benchmark methodology, we rank the software solutions.

© Springer International Publishing Switzerland 2015
G. Azzopardi and N. Petkov (Eds.): CAIP 2015, Part II, LNCS 9257, pp. 450–461, 2015.
DOI: 10.1007/978-3-319-23117-4_39

2 Related Work

For 3D reconstruction, various approaches such as *123D Catch* [3], *PhotoScan* [2], Photo tourism [23], VideoTrace [11], Kinect fusion [17], ProFORMA [18] etc. with various inputs like image collections, single images and video footage are in use. Each approach has its own drawbacks, for instance if stereo vision is used, depth information only up to a limited distance of typically less than $5m$ [12,20] is available for the reconstruction process. Furthermore 3D reconstructions have issues with e.g., shiny, textureless, or occluded surfaces, currently [19].

In archeology, traditional 3D recording technologies like terrestrial laser scanners or fringe projection systems, are still expensive, inflexible and often require expert knowledge for operation. Hence, most benchmarks, evaluation and taxonomies of 3D reconstruction software is published in the archeology context. Kersten and Lindstaedt [13] demonstrated and discussed the application of image based reconstruction of archaeological findings. Additionally, the accuracy of data produced by multi-image 3D reconstruction software is compared to the traditional terrestrial 3D laser scanners and light detection and ranging (LIDAR) systems [9,14].

3 Multi-view 3D Reconstruction

In this chapter we describe our benchmark as well as the chosen software solutions and data sets.

3.1 Multi-view Software Solutions

While there is a large body of academic and commercial software solutions for 3D reconstruction out of multi-view data sets, we chose four most well-known ones for our evaluation: *Agisoft PhotoScan Standard Edition* [2], *Autodesk 123D Catch* [3], *VisualSFM* [30–32] with *CMVS* [10] and *ARC 3D* [28]. With respect to other software solutions [1,15], these four tools are, in our opinion, the most widely used. Moreover these four tools are constantly present in many articles [9, 13,14,22].

PhotoScan Standard Edition [2] is introduced as first software. It is the only fee-based software solution in this benchmark, but provides quite a lot of features like photogrammetric triangulation, dense point cloud generation and editing, 3D model generation and texturing, and spherical panorama stitching. *PhotoScan* is available for *Windows*, *Mac OS* and *Linux* distribution and supports GPU acceleration during 3D reconstruction.

123D Catch by *Autodesk* [3] creates 3D models from a collection of up to 70 images. Currently this software is a free solution and is available for *Windows*, *Mac OS* and *Android*. To increase speed during the overall process, the reconstruction process is outsourced to cloud computing, thus an internet connection for uploading the images is needed. Since it is a software for users without

expert knowledge, only a few parameters can be set. As a result, the reconstruction process is quite intuitive.

Wu's *VisualSFM* is an academic software solution, which bundles his SFM-approaches [30–32] with Furukawa et al.'s multi-view-stereo techniques [10] into a powerful tool. As an academic software it is freely available for *Windows*, *Mac OS* and *Linux*. Like *PhotoScan* , it uses GPU acceleration, and especially for *nVidia* graphic cards CUDA is supported.

ARC 3D is an academic web-based 3D reconstruction service primarily designed for the needs of the cultural heritage field. Due to its web-based design the automatic reconstruction process including preprocessing steps like feature point detection, set of image pairs computation, camera calibration and full scale reconstruction is running on a cluster of computers at the Departement of Electrical Engineering of the K.U.Leuven. With the *ARC 3D* upload tool the user uploads a photo sequence over the Internet to the cluster. When the cluster has finished the reconstruction process, the user is notified by email and can download the results. A closer look into *ARC 3D* is given by Vergauwen and Van Gool [28].

3.2 Data Sets

During the planning stage of this benchmark, very soon it became obvious that a benchmark should include real scene photographs as well as photographs taken in a controlled indoor environment. Another essential requirement to the multi-view data sets is the availability of a ground truth. Based on these two criteria, several multi-view data sets were examined [6,21,24,25,29] and the below mentioned ones are chosen.

Since ground truth is required, the data sets *fountain-P11* and *Herz-Jesu-P8* [25] as a real scene and the *Oxford Dinosaur* [29] as planned photographs are chosen. The data set *dino* and *temple* [21] as planned photographs are also considered, but due to scaling problems our first submission to the evaluation platform did not succeed. These results will be integrated into future work. Data sets for special issues as, e.g., repeated structures [6] are not taken into account, because they can cause anomalies in the reconstruction pipelines of the software solutions.

We now describe the used data sets — the *fountain-P11*, *Herz-Jesu-P8* and *Oxford Dinosaur* — and the necessary adaptations made in this paper in detail. The first two pictures of Figure 1 show examples of the multi-view data set of outdoor architectural heritage by Strecha et al. [25], initially designed for benchmarking automatic reconstruction from multiple view imagery as a replacement for LIDAR systems. The scenes of the fountain and the Herz-Jesu church have been captured with a Canon D60 with a resolution of 3072×2028 pixels. The data set comprises eleven images of the fountain and eight of the Herz-Jesu church. The corresponding ground truth has been taken by a LIDAR system. A more detailed description of these data sets including the estimation procedures can be found in [25].

As a data set for a controlled indoor environment, the quite old *Oxford Dinosaur* [29] data is used. This is because such toy models, as seen in the third picture of Figure 1, are quite interesting for 3D printing. This data set includes 36 images of 720×576 pixels of a toy dinosaur captured on a turntable. As there is no ground truth available, the meshed model of Fitzgibbon et al. [8] results is taken as ground truth.

In order to provide the same conditions for all reconstruction solutions, the data sets must be adapted to a data format that all reconstruction tools can handle. Thus, all images have been converted to JPEG. As JPEG is a lossy compression algorithm, some details of the native data sets get lost. In the preparation phase, the data set *Oxford Dinosaur* causes some problems like incomplete reconstruction. To improve this behavior, we decided to also provide this data set with removed black background as seen in the last picture of Figure 1.

3.3 3D Model Generation

For 3D model generation, the converted images are put into the processing pipeline of each above mentioned solution.[1] Other additional a priori information, e.g., on internal camera calibration parameters was not provided for the reconstruction process. On a stand alone computer system dense point clouds for the data sets were computed and exported sequentially with all software solutions. The system was equipped with a 4-Core *Intel i7-3770* at 3.4*Ghz*, 16*GB* of RAM and a *nVidia Quadro K600* graphics card running *Windows 7 64bit* as operating system. Using different parameter configurations in the solutions, the user has a limited influence on the resulting dense point cloud. To simplify the benchmark, the initial default parameters of each software tool are taken if the model creation succeeded. For the reproduction of this benchmark, the parameters used are attached in Table 2. As output format the polygon file format (ply) is selected. As *123D Catch* cannot export the model as ply, the model was converted from obj to ply by Meshlab [16]. The resulting reconstructed 3D models of all software tools are shown in the center of Figure 1. Unfortunately, *ARC 3D* was not able to deliver results for the *Oxford Dinosaur*, neither with the original set nor with the background-removed set.

4 Comparison and Evaluation

The trickiest part is to define an evaluation scheme to compare the reconstructed models of Section 3.3. Each model comprises a particular number of points, and some of the reconstructed models have points beyond the boundaries of the ground truth. Hence, a simple point to ground truth comparison can generate poor results.

[1] For the benchmark *PhotoScan* Version 1.1.0.2004, *123D Catch* Build 3.0.0.54, *VisualSFM* Version 0.5.26 and *ARC 3D* uploader Version 2.2 was used.

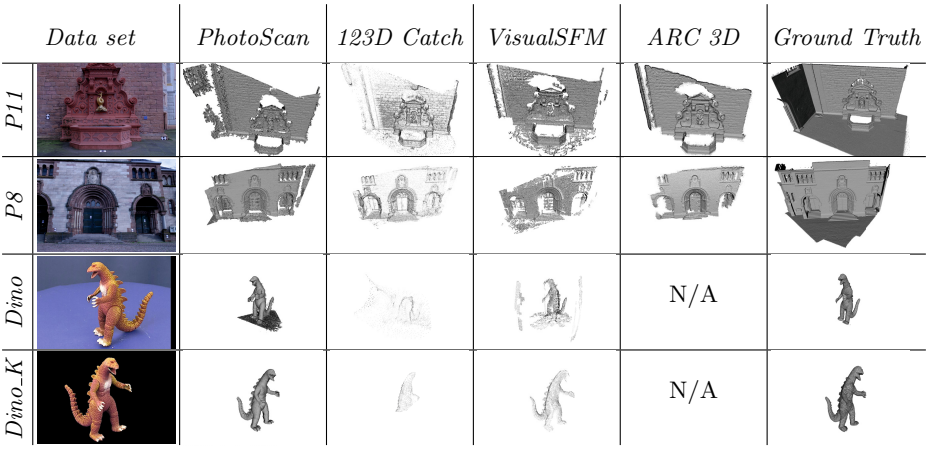


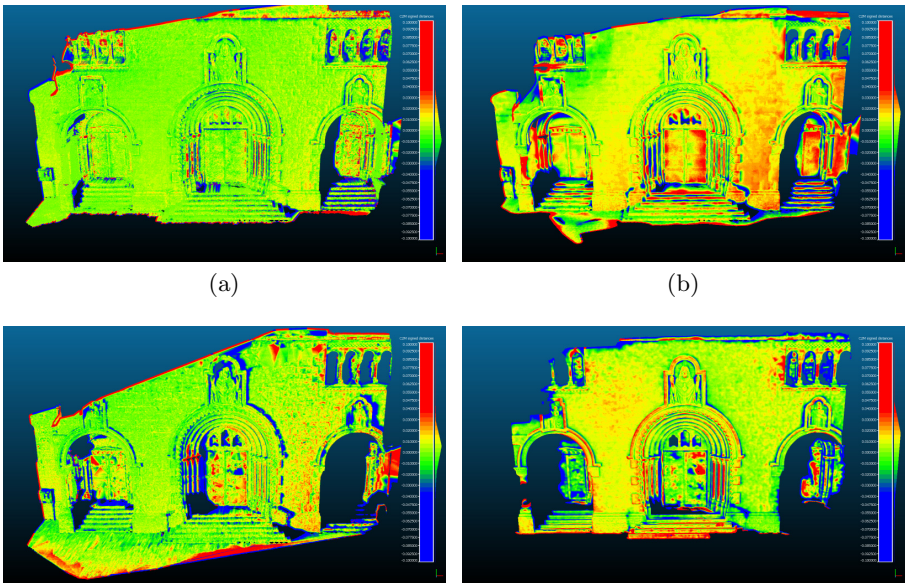
Fig. 1. Created 3D models of *PhotoScan* [2], *123D Catch* [3], *VisualSFM* [30–32] and *ARC 3D* [28] on four data sets: *fountain-P11* (11 images), *Herz-Jesu-P8* (8 images), *Oxford Dinosaur* (36 images) and *Oxford Dinosaur* with removed background (36 images). One sample image of each data set is shown on the left, reconstruction results of each tool are shown in the center and the corresponding ground truths are shown on the right.

4.1 Methodology

For this benchmark we mainly focus on the accuracy of the reconstructed models. The least common denominator of all the software involved is a dense point cloud of the model, which is used as starting point for our comparison, if no sufficient triangular mesh is provided by the software. We also documented the computing time required by each tool. But due to the web-based architecture of *123D Catch* and *ARC 3D*, the runtime does not give considerable evidence about the computational effort.

The following 3D model comparison pipeline includes the open source software *Meshlab* [16] and *CloudCompare* [7] for all 3D operation as well as *Matlab* for provision of statistics.[2] Based on the dense point clouds of each model, a rough direction and size alignment with the ground truth date is performed manually. Thereby, the global coordinate system is scaled on meters. If no mesh is provided, the aligned point clouds are meshed with the ball-pivoting algorithm by Bernardini et al. [4]. As pivoting ball radius the auto-guess setting of *Meshlab* is used. At this stage, the ground truth data alongside with each model is loaded into *CloudCompare*. In order to finely register the model with the ground truth, the iterative closest point algorithm (IPC) by Besl and McKay [5] with a target error difference of $1 \cdot 10^{-8}$ is used. For the registered models the minimal distance between every point to any triangular face of the meshed model is computed. Using the normal of the closest face the sign of each distance value is set. Note, that the comparison is made between each created, meshed model to the point

[2] Used 64*bit* version: *Meshlab* 1.3.3, *CloudCompare* 2.6.0, *Matlab R*2014.



(a) (b) (c) (d)

Fig. 2. Heat map of the minimal distance between the ground truth point cloud and the triangular mesh of the created model as reference on the *Herz-Jesu-P8* data set. Points with distance differences more than ±0.1 are not visualized; distance differences between -0.05 and 0.05 are colorized in the scheme blue–green–red. On the right next to the legend, the distance distribution is shown. Models are created by (a) *PhotoScan*, (b) *123D Catch*, (c) *VisualSFM* and (d) *ARC 3D*.

cloud of the ground truth, so the created model is set as the reference. For qualitative results, all distances of more than ±0.1 are not visualized, an example is shown in Figure 2. Distances between -0.1 to -0.05 are colorized blue, red is used for 0.05 to 0.1. Between -0.05 and 0.05 the color scheme blue–green–red is applied.[3] The scaling values for the *Oxford Dinosaur* data sets differ due to smaller model sizes. For quantitative results, all computed distances of a model are exported. On these exported data the mean value (μ) and standard deviation (σ) of the distance distribution as seen as in Table 1 are calculated. Further, Figure 3 shows the histograms of each model from each tool to represent the distance distribution. For a direct comparison, an empirical cumulative distribution function (CDF) is calculated and plotted in Figure 4. The computation time of each solution is simply measured in seconds and can be found in Table 1.

4.2 Reasoning of Our Methodology

In contrast to common practice [26,27], we made the comparison between each reconstructed model as the reference to the ground truth, and not vice versa.

[3] For more results cf. https://ikw.uos.de/~cv/publications/caip15/

Why? Each software solution yields a different number of dense points, partially depending on parameter settings, which can not be influenced by the user. To get comparable histograms and statistic figures, the same number of points for each model is needed. Since it is not possible to parameterize all solutions such that all reconstructed models have the same number of points, random sampling from the points appears to be a good idea at first glance. However, a random sample of points may cause misleading results in our setup, because some reconstructed models protrude beyond the ground truth. For that reason we do not use random sampling. Our method to compare the ground truth to the created models (as reference) is much simpler, by this means, we always have the same numbers of points.

A second particularity of our methodology could be the comparison against the meshed model and not against the point cloud. The fact that the different software solutions provide a different number of points leads in most cases to substantial distances, if only a small number of points is available. Thus, the distance is calculated to the mesh to not adversely affect tools which creates a small number of points.

5 Result and Discussion

Three of the four tools were able to compute models out of the four data sets as seen in Figure 1. Further, this figure shows that all tested software tools yield useful results for data sets *fountain-P11* and *Herz-Jesu-P8*. However, for the two *Oxford Dinosaur* data sets, only *PhotoScan* and *VisualSFM* come up with useful results, the result of *123D Catch* is seriously deformed or incomplete and *ARC 3D* returns a *ARC3D reconstruction failed* email with no model. We made a qualitative ranking based on the heat maps of distances, see Figure 2 as examples. The heat maps make clear that *PhotoScan* exhibits exceedingly few deviations to the ground truth, followed by *VisualSFM*, *ARC 3D* and *123D Catch*. However, the ranking of *ARC 3D* is done bearing in mind it has failed on two data sets. A printable model of the toy dinosaur from the *Oxford Dinosaur* has only been created by *PhotoScan* and *VisualSFM*.

For the quantitative analysis we excluded *ARC 3D* for the reason of the missing models. As seen in Figure 3 we assume a statistical normal distribution of the distance deviation. The quantitative ranking is done by a scoring system. On each data value the best value gets one and the worst gets three scoring points. The mean value (μ), the standard deviation (σ) and the time are scored separately. For example, the model created by *VisualSFM* provides the lowest mean value deviation, which is indicated in Table 1 by the lowest average of points for mean value. Finally, the quantitative ranking based on the mean value (μ) and standard deviation (σ) is headed by *VisualSFM* (12 points) followed by *PhotoScan* (17 points) and *123D Catch* (19 points). To confirm the previous ranking and to rank *ARC 3D* we also analyse the CDF. As seen in all plots of Figure 4, the probabilities are mainly close to zero. The probability distribution in the CDF reflects the results of Table 1. Neglecting the data set of the *Oxford Dinosaur*, *ARC 3D* can be inserted between *PhotoScan* and *123D Catch*.

Table 1. Mean value (μ), standard deviation (σ) and computation time of each software tool with the scoring in points on each attribute.

	PhotoScan [2]			*123D Catch* [3]			*VisualSFM* [30–32]			*ARC 3D* [28]		
	μ[m]	σ[m]	time[s]	μ[m]	σ[m]	time[s]	μ[m]	σ[m]	time[s]	μ[m]	σ[m]	time[s]
fountain-P11	$-1.90 \cdot 10^{-2}$	1.01	9136	$1.87 \cdot 10^{-1}$	$83.73 \cdot 10^{-1}$	600	$9.14 \cdot 10^{-3}$	1.14	221	$-2.14 \cdot 10^{-2}$	1.47	1620
Herz-Jesu-P8	$-8.59 \cdot 10^{-2}$	1.37	1634	$9.95 \cdot 10^{-2}$	1.36	900	$-4.58 \cdot 10^{-3}$	1.31	170	$-2.56 \cdot 10^{-1}$	2.05	1380
Dinosaur	$6.09 \cdot 10^{-4}$	$2.65 \cdot 10^{-3}$	139	$8.44 \cdot 10^{-3}$	$2.01 \cdot 10^{-2}$	420	$5.43 \cdot 10^{-4}$	$4.37 \cdot 10^{-4}$	30	—	—	—
Dinosaur_K	$4.52 \cdot 10^{-4}$	$2.48 \cdot 10^{-3}$	106	$3.86 \cdot 10^{-5}$	$2.06 \cdot 10^{-2}$	720	$-1.55 \cdot 10^{-4}$	$3.62 \cdot 10^{-3}$	28	—	—	—
Scoring	9	8	10	10	9	10	5	7	4	—	—	—

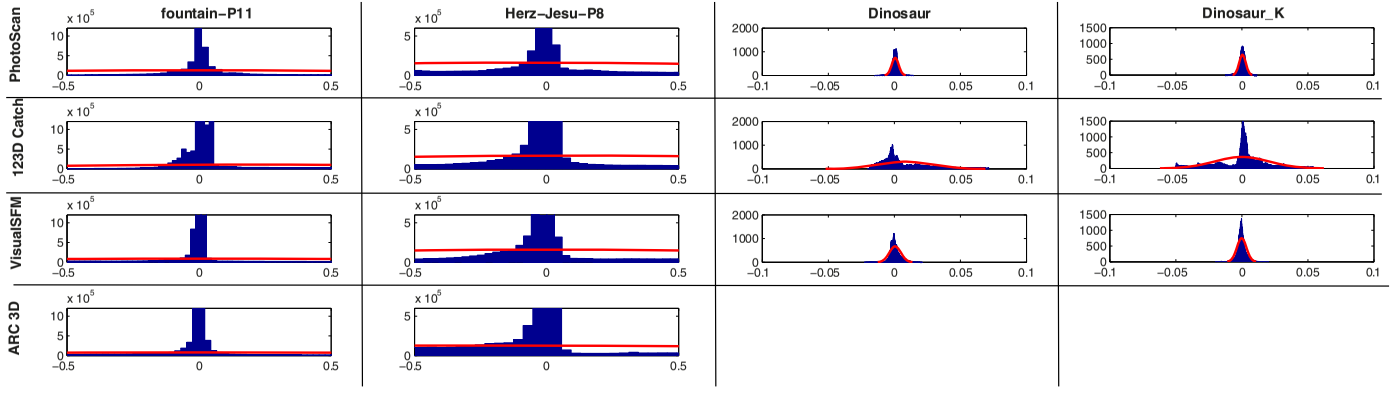


Fig. 3. Histogram plots of the distances between the created models and the ground truth data. *fountain-P11* contains 12,991,849, *Herz-Jesu-P8* contains 18,101,559 and *Oxford Dinosaur* with our without background contains 67,448 elements. The normal distribution fit is represented by the red line.

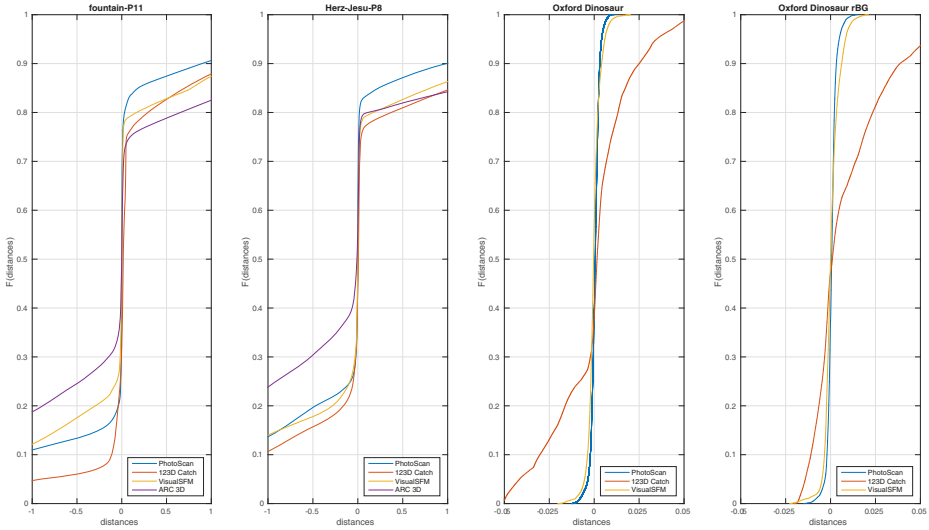


Fig. 4. Empirical cumulative distribution function per data sets. It describes the probability of the distance between the created model and the ground truth normalized on a scale from 0 to 1 over the distance. $F(distance) = P(-a < distance < a)$ with $a = 1$ for *fountain-P11* and *Herz-Jesu-P8* or with $a = 0.05$ for both *Oxford Dinosaur* sets

Based on this benchmark data, we deduce our own ranking, which also considers soft facts like runtime, license and execution by command line. As a result, *VisualSFM* is our selection of choice because of its quality, academic licensing and runtime. In our ranking, *VisualSFM* is followed by *PhotoScan*, *ARC 3D* and *123D Catch*.

In consideration of our benchmark, several issues can be questioned. First, we have noticed the used ground truth model of the *Oxford Dinosaur* does not seem to represent the real proportions, it could be stretched in height. Furthermore, problematic reconstruction scenes with repeated, shiny or textureless structures have been purposely neglected.

6 Conclusion

This paper presents a benchmark which compares the performance of multi-view 3D reconstruction software qualitatively and quantitatively on four different data sets. The data sets are two outdoor architectural scenes and planned shots of a toy dinosaur. Four different software solutions — *PhotoScan*, *123D Catch*, *VisualSFM* and *ARC 3D*— were tested and ranked. Summarizing, the quality of the tested tools is remarkably good but focused on specific fields. The available software solutions for multi-view 3D reconstruction can be ranked in terms of reconstruction quality, runtime etc. but due to the broad application field it is not possible to provide a general ranking.

Acknowledgments. This work was funded by German Research Foundation (DFG) as part of the Priority Program "Scalable Visual Analytics" (SPP 1335).

All *ARC 3D* marked 3D models are created by the *ARC 3D* webservice, developed by the VISICS research group of the KULeuven in Belgium.

References

1. Acute3D: Smart3DCapture - Acute3D — capturing reality with automatic 3D photogrammetry software, January 2015. http://www.acute3d.com/
2. Agisoft: Agisoft PhotoScan, January 2015. http://www.agisoft.ru/
3. Autodesk Inc: Autodesk 123D Catch — 3D model from photos, January 2015. http://www.123dapp.com/catch
4. Bernardini, F., Mittleman, J., Rushmeier, H., Silva, C., Taubin, G.: The ball-pivoting algorithm for surface reconstruction. IEEE Trans. Vis. Comput. Graphics **5**(4), 349–359 (1999)
5. Besl, P., McKay, H.: A method for registration of 3-D shapes. IEEE Trans. Pattern Anal. Mach. Intell. **14**(2), 239–256 (1992)
6. Ceylan, D., Mitra, N.J., Zheng, Y., Pauly, M.: Coupled structure-from-motion and 3D symmetry detection for urban facades. ACM Trans. Graphics **33**(1), 1–15 (2014)
7. CloudCompare : Cloudcompare, January 2015. http://www.danielgm.net/cc/
8. Fitzgibbon, A.W., Cross, G., Zisserman, A.: Automatic 3D model construction for turn-table sequences. In: Koch, R., Van Gool, L. (eds.) SMILE 1998. LNCS, vol. 1506, pp. 155–170. Springer, Heidelberg (1998)
9. Fonstad, M.A., Dietrich, J.T., Courville, B.C., Jensen, J.L., Carbonneau, P.E.: Topographic structure from motion: a new development in photogrammetric measurement. Earth Surface Processes and Landforms **38**(4), 421–430 (2013)
10. Furukawa, Y., Curless, B., Seitz, S.M., Szeliski, R.: Towards internet-scale multi-view stereo. In: IEEE Computer Vision and Pattern Recognition (CVPR), pp. 1434–1441 (2010)
11. van den Hengel, A., Dick, A., Thormählen, T., Ward, B., Torr, P.H.S.: VideoTrace. ACM Trans. Graphics **26**(3), 86 (2007)
12. Henry, P., Krainin, M., Herbst, E., Ren, X., Fox, D.: RGB-D mapping: using depth cameras for dense 3D modeling of indoor environments. In: Khatib, O., Kumar, V., Sukhatme, G. (eds.) Experimental Robotics. STAR, vol. 79, pp. 477–491. Springer, Heidelberg (2012)
13. Kersten, T.P., Lindstaedt, M.: Image-based low-cost systems for automatic 3D recording and modelling of archaeological finds and objects. In: Ioannides, M., Fritsch, D., Leissner, J., Davies, R., Remondino, F., Caffo, R. (eds.) EuroMed 2012. LNCS, vol. 7616, pp. 1–10. Springer, Heidelberg (2012)
14. Koutsoudis, A., Vidmar, B., Ioannakis, G., Arnaoutoglou, F., Pavlidis, G., Chamzas, C.: Multi-image 3D reconstruction data evaluation. Journal of Cultural Heritage **15**(1), 73–79 (2014)
15. Lukas Mach: insight3d - opensource image based 3D modeling software, January 2015. http://insight3d.sourceforge.net/
16. MeshLab: Meshlab, January 2015. http://meshlab.sourceforge.net/

17. Newcombe, R.A., Izadi, S., Hilliges, O., Molyneaux, D., Kim, D., Davison, A.J., Kohi, P., Shotton, J., Hodges, S., Fitzgibbon, A.: Kinectfusion: real-time dense surface mapping and tracking. In: IEEE International Symposium on Mixed and Augmented Reality (ISMAR), pp. 127–136 (2011)
18. Pan, Q., Reitmayr, G., Drummond, T.: Proforma: probabilistic feature-based online rapid model acquisition. In: Proceedings of the British Machine Vision Conference (BMVC) (2009)
19. Schöning, J., Heidemann, G.: Interactive 3D modeling - a survey-based perspective on interactive 3D reconstruction. In: Proceedings of the International Conference on Pattern Recognition Applications and Methods (ICPRAM), vol. 2, pp. 289–294 (2015)
20. Schöning, J., Pardowitz, M., Heidemann, G.: Taxonomy of 3D sensors: A survey of state-of-the-art low cost 3D-reconstruction sensors and their field of application. Low Cost 3D (2014)
21. Seitz, S., Curless, B., Diebel, J., Scharstein, D., Szeliski, R.: A comparison and evaluation of multi-view stereo reconstruction algorithms, vol. 1, pp. 519–528 (2006)
22. Singh, S.P., Jain, K., Mandla, V.R.: 3D scene reconstruction from video camera for virtual 3d city modeling. American Journal of Engineering Research **3**(1), 140–148 (2014)
23. Snavely, N., Seitz, S.M., Szeliski, R.: Photo tourism: Exploring photo collections in 3d. ACM Trans. Graphics **25**(3), 835 (2006)
24. Société Internationale de Photogrammétrie et de Télédétection (ISPRS): The isprs data set collection, January 2015. http://www.isprs.org/data/default.aspx
25. Strecha, C., von Hansen, W., Van Gool, L., Fua, P., Thoennessen, U.: On benchmarking camera calibration and multi-view stereo for high resolution imagery. In: IEEE Computer Vision and Pattern Recognition (CVPR), pp. 1–8 (2008)
26. Thoeni, K., Giacomini, A., Murtagh, R., Kniest, E.: A comparison of multi-view 3D reconstruction of a rock wall using several cameras and a laser scanner. Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. XL-5, 573–580 (2014)
27. Uh, Y., Matsushita, Y., Byun, H.: Efficient multiview stereo by random-search and propagation. In: International Conference on 3D Vision (2014)
28. Vergauwen, M., Van Gool, L.: Web-based 3D reconstruction service. Machine Vision and Applications **17**(6), 411–426 (2006)
29. Visual Geometry Group, University of Oxford: Multi-view and oxford colleges building reconstruction - dinosaur, August 2014
30. Wu, C.: Towards linear-time incremental structure from motion. In: International Conference on 3D Vision (2013)
31. Wu, C.: SiftGPU: A GPU implementation of scale invariant feature transform (SIFT), January 2015. http://cs.unc.edu/ccwu/siftgpu
32. Wu, C., Agarwal, S., Curless, B., Seitz, S.M.: Multicore bundle adjustment. In: IEEE Computer Vision and Pattern Recognition (CVPR) (2011)

Appendix

Table 2. Software parameters used for the reconstruction process

	Parameters
PhotoScan	*Align Photos* Accuracy: High, Pair preselection: Disable, Key point limit:40000, Tie point limit:1000 *Build Dense Cloud* Quality: Ultra high, Depth filtering: Aggressive
123D Catch	*Mesh Quality* Maximum
VisualSFM	Default settings
ARC 3D	*Subsample images* Subsample to (%): 100