

Benchmarking Structure From Motion Algorithms of Urban Environments with Applications to Reconnaissance in Search and Rescue Scenarios

Angel Martell¹, Helge A. Lauterbach², Klaus Schilling^{1,2} and Andreas Nüchter^{1,2}

Abstract—Structure from motion is a common approach to generate 3D models of objects and structures. The ease of the data acquisition and the wide selection of available algorithms makes the technique easily accessible. Previous benchmarks on the topic have been focused on scanning small structures, specially for archaeology, or have been limited to single or very few algorithms. In this work different algorithms are benchmarked regarding accuracy and processing time for datasets acquired in urban environments, with the goal of analyzing the feasibility of utilizing this technique on search and rescue operations. Real-world rescue scenarios are demanding due to the presence of challenging surfaces and smoke.

I. INTRODUCTION

Surveying of search and rescue (SaR) scenarios usually involves manual inspection by rescuers, and is a dangerous and time consuming process. As the survival of victims depends on the time it takes for being rescued, research in new technologies that can automate this task without risking human lives is of utter importance. Currently, this task is aided with additional surveying by aerial photography and also thermography taken from an helicopter, but the data inspection for this methods is complex, as there is no spatial information available for analysis. Structure from motion (SfM) is a technique in the field of photogrammetry that computes spacial information based only on image data. A commonly available UAV with an integrated camera is therefore sufficient to capture the required data. Due to the low cost of the equipment used and the ease of acquiring the information, it serves as a possibility to be used in SaR scenarios. In this work we present benchmarks of SfM reconstructions of urban scenarios using different available softwares, and analyze the feasibility of this technique, to be used in SaR situations.

II. RELATED WORK

Since SfM is a recent and constantly improving topic, very few research has been done in asserting the accuracy of different algorithms. At the workshop "Multiview Relationships in 3D Data" at ICCV 2017, Vladlen Koltun encouraged researchers to provide benchmarks for 3D reconstruction algorithms. In a previous benchmark, he and

others tested several SfM algorithm pipelines. They used the harmonic average of the precision and recall (F-score) of the models error compared to a laser scanner reference, as to rank the outputs in both accuracy and completeness [1].

Most comparisons between different SfM algorithms have been made in close range measurements of small objects, like ornaments or sculptures. Nikolov et al. [2] tested six different commercial SfM software packages, digitalizing six objects with varying characteristics. Most algorithms provided sub-millimeter accuracy for close range photos but failed to resolve finer details due to oversmoothing.

Jaud et al. [3] compared AgiSoft PhotoScan [4] against MicMac [5], by building a model of a landslide on the Reunion island with images taken from an hexacopter. The generated models are compared to laser scanner measurements, resulting in average errors between 3 - 4 cm. In steep slopes the errors rise up to 17 cm.

Similarly, Pangajotidis et al. [6] compared a reconstruction of tree stems using PhotoScan and compared to a reference frame from terrestrial laser scans. They found errors of up to 11 cm.

Mancini et al. [7] and Genchi et al. [8] also used UAVs for capturing data and generating models for topography. Comparison to ground truth results in an accuracy of 19 cm for PhotoScan and 7 cm for Visual SfM respectively.

Our benchmark focuses on reconstruction of urban structures including accuracy and runtime evaluation of eight SfM pipelines. The remainder of this paper is organized as follows. Section III gives an overview of the benchmarked algorithms. Section IV-A and IV-B describe the experimental setup and evaluation process followed by a discussion of the results in Section IV-C.

III. BENCHMARKED ALGORITHMS

The general pipeline steps for SfM reconstruction are depicted in Fig. 1. At first, feature detection is performed. The majority of algorithms mentioned below use SIFT feature detection. After the features are identified on every image, every possible pair of images on the dataset is processed to find matching features between the two images on the pair. If a coherent match is found, then the pair is added as input for the next step. In the sparse reconstruction step the relative camera poses and orientations are estimated from the computed pairwise matches and aggregated with other pairs of images to build a 3D model of the photographed object. This aggregation is either sequential, where the errors are

¹ Center for Telematics, 97074 Würzburg, Germany, angel.martell@telematik-zentrum.de

² Department of Informatics VII — Robotics and Telematics, Julius-Maximilians-University Würzburg, Würzburg, Germany {helge.lauterbach|andreas.nuechter}@uni-wuerzburg.de

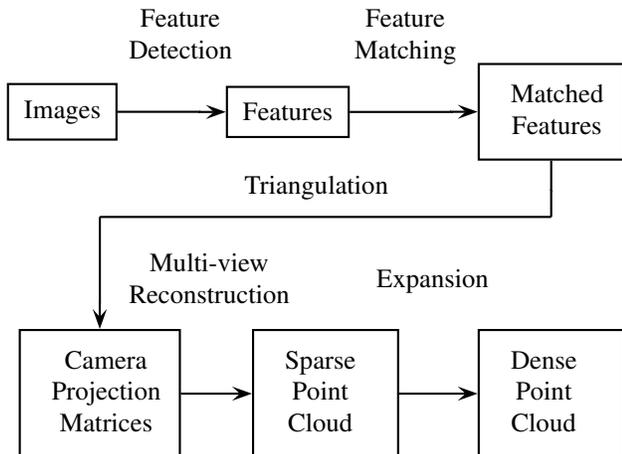


Fig. 1: Structure from motion pipeline steps

minimized via a bundle adjustment algorithm as pairs are added to the model; or global, where errors are minimized by matching all pairs at once. Since the global approach has very strict requirements to work properly (all features must be visible by all cameras, for example), all SfM algorithms implement the sequential sparse reconstruction approach. The output of the sparse reconstruction stage is a 3D model of the object formed by the detected features, and it also includes the pose (location and orientation) of the cameras. As a final stage, a densification stage follows in which the goal is to assign spatial 3D coordinates to all or most of the pixels in the input images, as to increase the count of points in the 3D model and include color information [9].

In this work, we focus on the reconstruction steps up to dense reconstruction, i.e., meshing is not considered.

a) Visual Structure from Motion: has been released in 2013 by Wu Changchang [10]. The complexity of the reconstruction problem is reduced by preemptive feature matching where only the h largest scales features are used to decide if a pair contains matches or not before using a complete feature match, acceleration of incremental multi-view reconstruction by replacing matrix inversion with implicit estimation; and doing bundle adjustment only for a subset of the latest n cameras added to the model, and not to the complete model. For dense reconstruction the Patch-based Multi-view Stereo toolchain from [11] is used.

b) Open Multiple View Geometry: released in 2012 by Moulon et al. [12] provides a pipeline for feature detection and matching and incremental structure from motion. In contrast to other SfM approaches, the camera pose estimation is improved by introducing adaptive thresholds for the model estimation instead of using globally fixed values.

c) Open Multiple View Stereovision: Dan Cernea et al. released a set of algorithms for dense reconstruction of SfM reconstruction. Based on the Patch-Match algorithm [13] the basic concept is to focus on regions that are more likely to have good matches based on the multi-view reconstruction.



Fig. 2: Aerial view of the practice hall at the Bavarian Firefighter School in Würzburg. The building features large white walls and reflective surfaces.

d) Multi-View Environment: provides a complete pipeline for image based geometry reconstruction [14]. It uses SIFT and SURF features jointly together which are matched pairwise and tracked over several views. The dense reconstruction based on [15] uses depth map estimation for each view and fusing them with the floating scale surface reconstruction algorithm [16].

e) COLMAP: was released in 2016 by Schönberger et al. [17]. Its main goal is to provide a general-purpose solution for structure from motion reconstruction that can work with high reliability under a variety of conditions. Improves the incremental structure from motion concept by including geometric verification to improve the triangulation. This includes exclusion of degenerated image configuration as panoramic image pairs and preference of image pairs with larger baselines

f) Agisoft PhotoScan [4]: is extensively used in research and industry [18], [19], [20], [21], [22], so we include the software in our evaluation.

IV. EXPERIMENTS AND RESULTS

A. Datasets and Preparation

Structure from motion algorithms have known limitations when reconstructing structures with texture-less surfaces, such as buildings with plain colour painted walls [23], as well with reflective surfaces and foliage. For evaluation we recorded two datasets at the Bavarian Firefighter School in Würzburg, Germany. The chosen buildings represent possible urban rescue scenarios.

A DJI Phantom 3 drone with a built-in camera was flown around the buildings in a "Point of Interest" flight mode to record video data with the camera tilted by approximately 45° . Reference data is provided by terrestrial laser scanning using a Riegl VZ 400 and scan registration with *3DTK — The 3D Toolkit* [24].

The first dataset includes the practice hall depicted in Fig. 2. The building is used to train firefighters in a variety of urban search and rescue scenarios. It houses a main street, office and urban environments, etc (indoor). The building is



Fig. 3: Aerial view of the burning house at the Bavarian Firefighter School in Würzburg. Light smoke is apparent at the chimneys and the front wall.

TABLE I: Recorded data per model

Model	Rec. Video	Laser scans	Num. of points
Practice hall	32.63 min	12	145M
Burning house	4.5 min	10	133M

a cuboid with a length of 76.8m, a width of 48.5m and a height of 40m. One of the sides features a large glass surface (Fig. 2 black front facade, equal to the top structure). This creates demanding reflections of surrounding structures. The building also features two large white walls to the sides without any prominent textures. For reference, which we will call ground truth, 12 terrestrial 3D laser scans were acquired around the building and on the main roof. The ceiling of the secondary structure was not accessible.

The second dataset, shown in figure 3, includes the so called burning house, that is used to provide realistic fire fighting training scenarios. Therefore smoke and fire are generated in a controllable environment. The structure resembles a typical German residential house with a footprint of $12.6\text{m} \times 12.5\text{m}$. Here, the ground truth point cloud consists of ten terrestrial laser scans.

The recorded datasets are available at <https://robotik.informatik.uni-wuerzburg.de/telematics/download/SSRR2018/>

For benchmarking SfM videos are recorded, images of size 2704×1520 are extracted with a rate of 1.2 fps (six frames every five seconds). For the second dataset, two more benchmarks are done: one reducing the rate of the extracted images to 0.6 fps (half-speed) and another one using the same rate as the first benchmark, but halving the width and height of the images (quarter-resolution). This is done to test how reducing the available input data affects the quality of the produced models. Table I gives details for the datasets.

B. Evaluation process

We evaluate eight different combinations of the algorithms presented in Section III. Besides running the complete pipeline within VisualSfM, MVE, COLMAP and Photoscan, the sparse point clouds from VisualSfM and openMVG are additionally used as input for the dense reconstruction of openMVS and MVE. The combinations are illustrated in Fig. 4. All algorithms are run with the parameter values

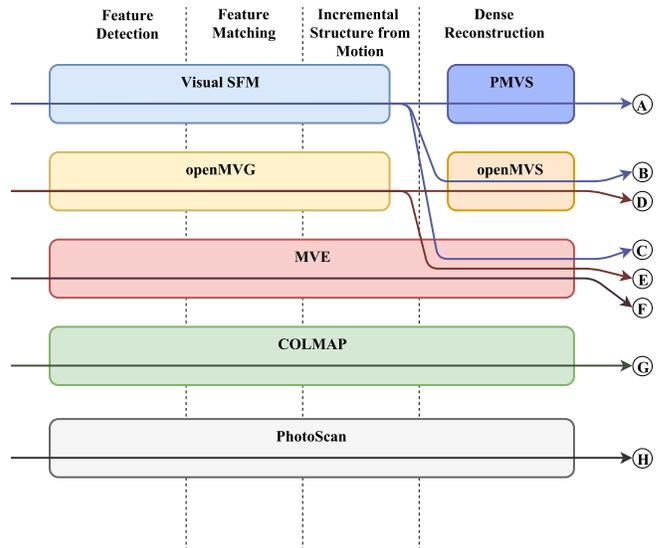


Fig. 4: Benchmarked Structure from Motion pipelines, inspired by [25].

they have defined as default. The camera intrinsic parameters are computed by automatic camera calibration during reconstruction, instead of being provided to the algorithms.

As the feature match stage in the typical structure from motion pipeline takes quadratic time to be executed as a function of the number of input images, a list indicating the pairs to be matched is provided to the algorithms. This simplification is done based on the assumption that comparing an image with the previous and next n images is appropriate, since they are taken from a continuous video and little overlap will occur on completely different segments of recordings. Also segments of video that have the same or similar features of the structure in view are also matched together, as to ensure loop closure. This is also true across multiple video files, as its images are also matched together across the different segments so they can be incremented to the same structure in the reconstruction stage. The same list with the matching pairs is provided to all pipelines.

As metrics for our benchmarks, the total runtime it takes the algorithm to complete the tasks and the cloud to cloud error distance compared to the laser scanner models will be used. Since the produced models via the SfM approach will have an unknown scale, these are matched to the laser scanner frame of reference using 3DTK using the unit quaternion minimization algorithm with scale adjustment [26]. Afterwards, the cloud to cloud error distance is computed with CloudCompare [27]. This software is also used to compute the expected error value and standard deviation are according to a Weibull distribution [28] that will be used to compare the results of the produced models.

a) *Execution time:* The data processing is to be done on an Intel Core i7-4700MQ with four cores operating at a frequency of 2.4 GHz and with Hyper-Threading enabled, and a nvidia GeForce GTX 765M GPU with 2 GB of video memory and 768 CUDA cores operating at a frequency of

850 MHz and 16 GB of memory. The computation times are recorded for each step of the pipelines.

b) Accuracy of point clouds: The generated models are aligned to the reference point cloud using 3DTK. Alongside registration the scale of each SfM models is determined. Accuracy of point clouds is evaluated by

- computing the cloud to cloud distance with CloudCompare.
- analyzing the error distribution by fitting the cloud to cloud error distance to a Weibull probability distribution with CloudCompare.

C. Results

A visual inspection of the generated models for the practice hall reveals that as expected, all algorithms showed problems in reconstruction of the white side walls of the building. However they were able to partially reconstruct the reflective surface, although major parts appear as additional structures inside the building.

Besides the regions where there was smoke in the recordings of the burning house, the models appear clean with very little noise. The influence of the smoke is minor except for those pipelines using MVE in dense reconstruction (Fig. 6c, 6e and 6f).

For both datasets the generated models of PhotoScan appear clean but compared to the other algorithms they are relatively sparse. It has the tendency to smoothen edges and smaller objects. This is especially noticeable at the corners of the roof of the practice hall as marked in Fig. 5h as well as the car in front of the burning house in Fig. 6h. On the other hand the spatial distribution of the reconstructed points is homogeneous.

The following findings are given for the practice hall dataset in Fig. 5:

- As expected all generated point clouds contain a lot of noise and artifacts due to reflections and white walls. Noise in the roof makes the secondary structure indistinguishable.
- Straight artifacts in multiple directions by MVE due to the reflections on the model.
- Failed to reconstruct side walls without textures.
- Most pipelines were able to reconstruct some part of the reflecting surface, except for PhotoScan.
- Color bleeding appears on edges of the models.

The following findings are given for the burning house dataset in Fig. 6:

- Color bleeding is not an issue for this dataset.
- Compared to Visual SfM, COLMAP manages to reconstruct texture-less surfaces better.
- Smoke seems to be not an influencing factor for most algorithms, except on where the MVE dense reconstruction algorithm is used.

a) Execution time: The Feature Detection step is linearly dependent on the number of images, so the execution times are comparably fast for all algorithms except for MVE. Feature Detection with MVE takes up to ten times longer

than with Visual SfM or COLMAP. One reason is, that MVE in contrast to the other methods additionally computes and jointly uses SURF features. Another reason is, that MVE does not use GPU parallelization, as the other algorithms do.

Feature Matching with PhotoScan takes the longest time for both datasets. In Contrast the preemptive feature matching approach of Visual SfM and COLMAP result in the fastest feature matching in this evaluation. However the speed up of COLMAP by using GPU parallelization in comparison to Visual SfM, that runs this step on the CPU is marginal. Note that the execution times for openMVG are even smaller. This is due fact, that openMVG only does preemptive feature matching in this stage, while the full feature matching is done in the sparse reconstruction. Since we provided a list of matching image pairs to all the algorithms, openMVG quickly terminates this step.

The simplifications in triangulation and bundle adjustment let Visual SfM out perform the other algorithms in the sparse reconstruction step with respect to the runtime. As openMVG computes the feature matches in this step, the sparse reconstruction takes a considerable large amount of time. However summing up the total run times up to this step, openMVG is faster than MVE and COLMAP for the burning house.

Regarding dense reconstruction, it was interestingly found, that MVE finishes fast when expanding its own sparse model, while it needs up to 14 times longer to densify models created by Visual SfM or openMVG. This is related to the fact, that the sparse models of the latter methods provide more points. In our experiments dense reconstruction with COLMAP takes a large amount of time. This is due to its two pass consistency checks. For the practice hall only PhotoScan is comparable slow, while producing less dense point clouds.

Regarding the total runtimes pipelines using Visual SfM are the fastest ones in this comparison as the incremental structure from motion step nearly runs in linear time. COLMAP and PhotoScan are the slowest pipelines. Their dense reconstruction consumes more time than the total runtime of the Visual SfM pipelines.

b) Accuracy of point clouds: The results of the evaluation to ground truth are given in Tab. II, when fitting the error data to a Weibull distribution. Fig. 5 visualizes the computed errors for the practice hall, Fig. 6 for the burning house respectively. Yellow indicates high point to point distances, darker colors represent low errors. The color scale goes from 0 m to 2 m.

The expected errors are in the order of magnitude of decimeters for both datasets, as previous studies have also found. The error is larger for the more complex structure of the practice hall, with the large reflecting surfaces affecting most of it. Also the standard deviation of the expected error is often larger than the error itself, indicating that the distribution is substantially spread to higher error values.

Comparing the pipelines, those using openMVG for sparse reconstruction produce the most accurate models for both datasets. High errors are present main in reflected areas of

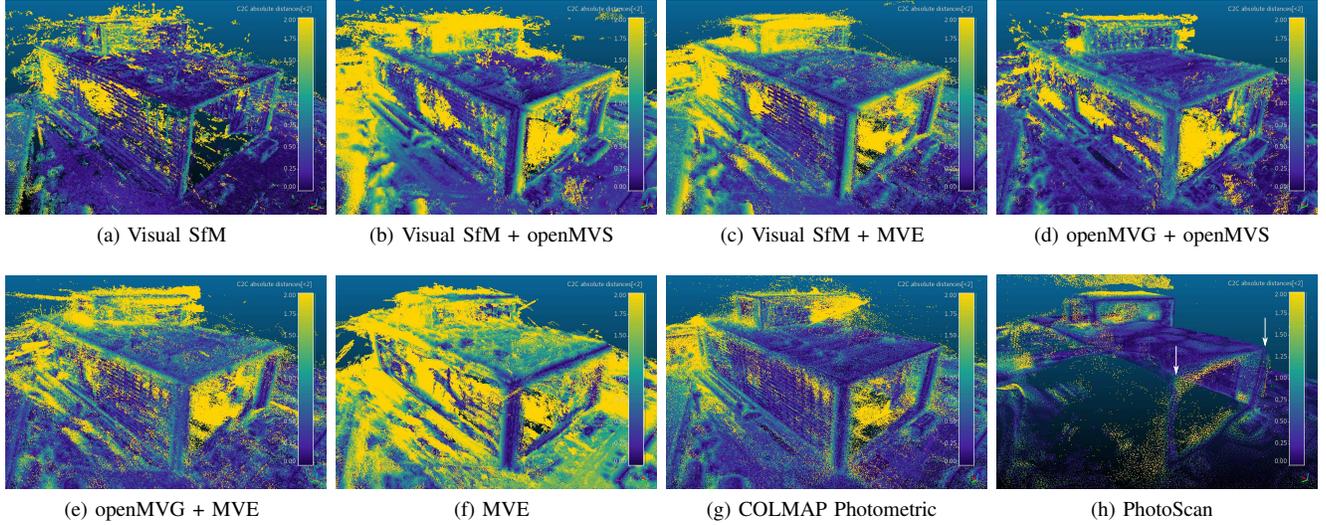


Fig. 5: Generated Point clouds for the practice hall colored by the distance to the reference cloud. The large reflective surface on the back of the building is also incomplete. The side wall of the building is completely missing.

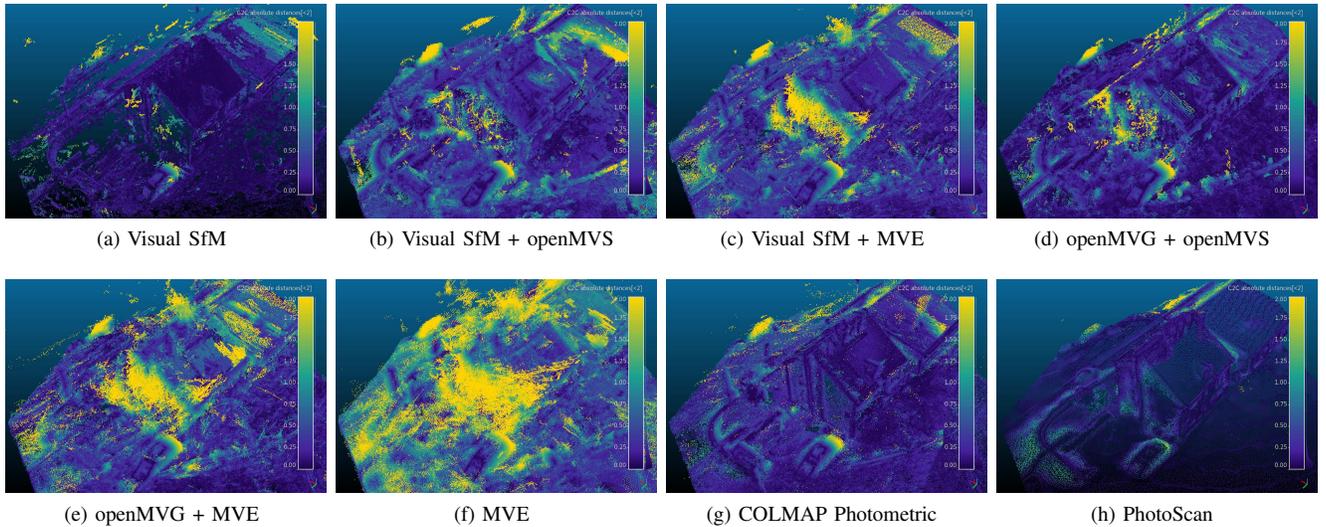


Fig. 6: Generated Point clouds for the burning house colored by the distance to the reference cloud. Algorithms where the expansion stage was made by MVE show the most noise in this view

TABLE II: Expected model to ground truth error and its standard deviation

Pipeline	Practice hall		Burning house		BH half speed			BH half resolution		
	E [m]	σ [m]	E [m]	σ [m]	E [m]	E Incr.	σ [m]	E [m]	E Incr.	σ [m]
Visual SfM	0.419	0.528	0.236	0.293	0.249	5.68%	0.099	0.258	9.52%	0.103
Visual SfM + openMVS	0.418	0.460	0.228	0.245	0.318	39.60%	0.128	0.320	40.50%	0.133
Visual SfM + MVE	0.320	0.388	0.141	0.142	0.241	71.51%	0.083	0.285	102.69%	0.116
openMVG + openMVS	0.260	0.394	0.116	0.129	0.173	49.60%	0.050	0.189	63.58%	0.054
openMVG + MVE	0.255	0.400	0.118	0.136	0.162	36.96%	0.044	0.221	87.25%	0.072
MVE	0.614	0.574	0.291	0.294	0.403	38.48%	0.188	0.399	37.04%	0.177
COLMAP Photometric	0.493	0.590	0.200	0.200	0.298	49.14%	0.131	0.340	69.90%	0.165
PhotoScan	0.597	0.680	0.225	0.260	0.375	66.66%	0.432	0.402	78.66%	0.452

TABLE III: Execution times in hours for Feature Detection (FD), Feature Matching (FM) and Sparse Reconstruction (SR)

Pipeline	Practice hall			Burning house			BH half speed			BH half resolution		
	FD	FM	SR	FD	FM	SR	FD	FM	SR	FD	FM	SR
Visual SfM	0.17	2.41	1.18	0.02	0.24	0.06	0.01	0.16	0.25	0.01	0.18	0.05
openMVG	0.34	0.17	18.15	0.05	0.02	0.25	0.02	0.01	0.07	0.01	0.01	0.06
MVE	0.89	10.94	1.22	0.12	0.17	0.34	0.02	0.09	0.07	0.03	0.17	0.24
COLMAP	0.10	1.99	6.09	0.02	0.20	0.45	0.01	0.12	0.15	0.01	0.13	0.28
PhotoScan	0.10	24.92	2.32	0.02	2.77	0.14	0.01	0.64	0.72	< 0.01	0.03	0.20

TABLE IV: Execution times in hours for Dense Reconstruction (DR), and total execution runtime for all pipelines

Pipeline	Practice hall		Burning house		BH half speed			BH half resolution		
	DR	Total	DR	Total	DR	Total	Reduc.	DR	Total	Reduc.
Visual SfM	5.00	8.76	0.31	0.63	0.25	0.66	-5.34%	0.13	0.37	41.77%
Visual SfM + openMVS	2.83	6.59	0.40	0.72	0.20	0.39	46.05%	0.40	0.64	11.31%
Visual SfM + MVE	3.88	7.63	2.17	2.49	0.96	1.15	53.66%	0.64	0.88	64.50%
openMVG + openMVS	2.42	21.08	0.39	0.71	0.19	0.28	60.53%	0.36	0.44	37.66%
openMVG + MVE	23.80	42.47	2.20	2.52	1.04	1.13	55.23%	0.56	0.64	74.62%
MVE	1.70	14.74	0.51	1.14	0.17	0.35	69.38%	0.13	0.57	49.96%
COLMAP Photometric	29.41	37.59	4.12	4.78	2.04	2.31	51.64%	4.04	4.47	6.54%
PhotoScan	37.73	65.08	1.67	4.61	0.31	1.68	63.55%	1.83	2.06	55.31%

the practice hall or areas with smoke in front of the burning house. Although Fig. 6e shows a lot of noise due to the smoke, the density in this particular area is lower than for the rest of the model and thus the expected error is less affected. The highest errors, consistently with the visual inspection, are found for MVE. Compared to its combination with openMVG the error is approximately 3 times higher.

The smoothed model of the practice hall built by PhotoScan shows similar high errors as MVE, however they are mostly found on the edges of the building. As the model is relatively sparse, the rounded edges have a high influence on the expected error. For the burning house the smoothing has smaller effects on the error, mainly because of the smaller dimensions of objects.

Note that in Fig. 5g and Fig. 6g, the generated models from COLMAP give the impression of more accurate models overall. Nevertheless Tab. II shows that the errors of the produced models with the openMVG + openMVS and openMVG + MVE pipelines are smaller than for COLMAP. This is because these pipelines generated a sparse point cloud of noise around the target structure that does not affect greatly the average error, but is dense enough to occlude the structure in the visualization made by CloudCompare.

c) Reducing amount of images and Resolution: In order to test the influence of using smaller datasets for gains in runtime, the pipelines for the burning house were rerun with a half speed extraction from the source video and quarter resolution. The resulting models visually do not significantly differ from the original models wrt. artefacts. However the error between the produced models and the laser reference increases by up to 100 percent, as Tab. II shows. In average, the error tends to increase by around 40 percent. On the other hand, the error deviation decreases with the smaller input dataset, indicating that the error is not as spread as with the larger input dataset.

A comparison of the pipeline groups using the same sparse reconstruction shows that the accuracy of the final models strongly depends on the dense reconstruction method. On one

hand, the accuracy for Visual SfM with its default dense reconstruction algorithm only slightly decreases when altering image amount and resolution respectively. On the other hand, the expected error increases by almost forty percent when using openMVS or MVE for dense reconstruction in Visual SfM. Comparable changes in accuracy are found for the remaining pipelines. In terms of accuracy loss, both variants are comparable with a slight advantage for the half speed variant.

Regarding runtime, the execution time for using only half of the input images is in average around fifty percent. Using quarter resolution the speed up is in general less. Due to the scale invariance of SIFT, the amount of features should be roughly the same, although with added noise. Although important speed gains were made, the execution time with reduced image data is still not in real time.

V. DISCUSSION, CONCLUSIONS AND FUTURE WORK

This work evaluated the generation of 3D models using SfM in rescue scenarios. A UAV with an integrated camera was used to acquire input images and the resulting 3D models were benchmarked against ground truth from laser scan data. The overall accuracy was in the range of decimeters.

A lesson learned from our experiments was that the high execution time needed to compute the models does not allow for any real-time application of the SfM yet, such as a quick disaster response data recollection method. It is usable as an auxiliary data recollection method though, as it is able to deliver results in less than one hour, if image count and size is kept low. It also allows for safely exploring a hard to reach area by ground without risking human lives. Most software pipelines were found to not being sensitive to moderate amounts of smoke in the scanned area.

As visual simultaneous localization and mapping (vSLAM) are emerging as methods that are able to run online, we expect them to be used in applications for rescue systems in the near future. In future work, we plan to replace the sparse reconstruction by vSLAM, to reduce run-time.

However, the majority of the computing time is currently spent in dense reconstruction.

Further future work will also concern UAV-based laser scanning methods to provide real-time 3D maps to rescue workers.

ACKNOWLEDGMENT

This work was partially funded by the project Eins3D (FZK 13N14183) from the Federal Ministry of Education and Research. We would like to thank Jürgen Schemmel from the Bavarian Firefighter School in Würzburg, Germany for the permission to record their buildings.

REFERENCES

- [1] A. Knapitsch, J. Park, Q.-Y. Zhou, and V. Koltun, "Tanks and temples: Benchmarking large-scale scene reconstruction," *ACM Trans. Graph.*, vol. 36, no. 4, pp. 78:1–78:13, July 2017.
- [2] I. Nikolov and C. Madsen, *Benchmarking Close-range Structure from Motion 3D Reconstruction Software Under Varying Capturing Conditions*. Cham: Springer International Publishing, 2016, pp. 15–26.
- [3] M. Jaud, S. Passot, R. L. Bivic, C. Delacourt, P. Grandjean, and N. L. Dantec, "Assessing the Accuracy of High Resolution Digital Surface Models Computed by PhotoScan and MicMac in Sub-Optimal Survey Conditions," *Remote Sensing*, vol. 8, no. 6, 2016.
- [4] AgiSoft. Photoscan. Last access on [2018-03-09]. [Online]. Available: <http://www.agisoft.com>
- [5] F. national geographic institute and F. national school for geographic sciences. Micmac. Last access on [2018-04-06]. [Online]. Available: <http://micmac.ensg.eu/index.php/Accueil>
- [6] D. Panagiotidis, P. Surový, K. Kuželka, *et al.*, "Accuracy of Structure from Motion models in comparison with terrestrial laser scanner for the analysis of DBH and height influence on error behaviour," *J. FOR. SCI.*, vol. 62, no. 8, pp. 357–365, 2016.
- [7] F. Mancini, M. Dubbini, M. Gattelli, F. Stecchi, S. Fabbri, and G. Gabbianelli, "Using Unmanned Aerial Vehicles (UAV) for High-Resolution Reconstruction of Topography: The Structure from Motion Approach on Coastal Environments," *Remote Sensing*, vol. 5, no. 12, pp. 6880–6898, 2013.
- [8] S. Genchi, A. Vitale, G. Perillo, and C. Delrieux, "Structure-from-Motion Approach for Characterization of Bioerosion Patterns Using UAV Imagery," *Sensors*, vol. 15, no. 2, pp. 3593–3609, 2015.
- [9] D. Robertson and R. Cipolla, *Practical Image Processing and Computer Vision*. John Wiley, 2009, ch. Structure from Motion.
- [10] C. Wu, "Towards Linear-Time Incremental Structure from Motion," in *Proceedings of the 2013 International Conference on 3D Vision*, ser. 3DV '13. Washington, DC, USA: IEEE Computer Society, 2013, pp. 127–134.
- [11] Y. Furukawa and J. Ponce, "Accurate, Dense, and Robust Multiview Stereopsis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 8, pp. 1362–1376, Aug. 2010.
- [12] P. Moulon, P. Monasse, and R. Marlet, "Adaptive Structure from Motion with a Contrario Model Estimation," in *Computer Vision – ACCV 2012: 11th Asian Conference on Computer Vision, Daejeon, Korea, November 5-9, 2012, Revised Selected Papers, Part IV*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 257–270.
- [13] C. Barnes, E. Shechtman, A. Finkelstein, and D. Goldman, "Patch-Match: A Randomized Correspondence Algorithm for Structural Image Editing," in *ACM SIGGRAPH 2009 Papers*, ser. SIGGRAPH '09. New York, NY, USA: ACM, 2009, pp. 24:1–24:11.
- [14] S. Fuhrmann, F. Langguth, and M. Goesele, "MVE: A Multi-view Reconstruction Environment," in *Proceedings of the Eurographics Workshop on Graphics and Cultural Heritage*, ser. GCH '14. Aire-la-Ville, Switzerland, Switzerland: Eurographics Association, 2014, pp. 11–18.
- [15] M. Goesele, N. Snavely, B. Curless, H. Hoppe, and S. Seitz, "Multi-View Stereo for Community Photo Collections," in *11th IEEE International Conference on Computer Vision (ICCV '07)*, 11 2007, pp. 1–8.
- [16] S. Fuhrmann and M. Goesele, "Floating Scale Surface Reconstruction," *ACM Trans. Graph.*, vol. 33, no. 4, pp. 46:1–46:11, July 2014.
- [17] J. Schönberger and J. Frahm, "Structure-from-Motion Revisited," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [18] F. Remondino, M. Spera, E. Nocerino, F. Menna, and F. Nex, "State of the art in high density image matching," *The Photogrammetric Record*, vol. 29, no. 146, pp. 144–166, 2014.
- [19] K. Thoeni, A. Giacomini, R. Murtagh, and E. Kniest, "A Comparison of Multi-view 3D Reconstruction of a Rock Wall using Several Cameras and a Laser Scanner," in *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XL-5, 06 2014, pp. 573–580.
- [20] G. Verhoeven, "Taking computer vision aloft archaeological three-dimensional reconstructions from aerial photographs with photoscan," *Archaeological Prospection*, vol. 18, no. 1, pp. 67–73, 2011.
- [21] S. Solb and R. Storvold, "Mapping Svalbard glaciers with the Cryowing UAS," in *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XL-1/W2, 09 2013.
- [22] P. Koppel, *Agisoft Photoscan: Point Cloud accuracy in close range configuration*. Koppel Engineering, 2015.
- [23] P. Saponaro, S. Sorensen, S. Rhein, A. Mahoney, and C. Kambhmettu, "Reconstruction of textureless regions using structure from motion and image-based interpolation," in *2014 IEEE International Conference on Image Processing, ICIP 2014*, 01 2015, pp. 1847–1851.
- [24] A. Nüchter. (2018) 3DTK - the 3D toolkit. Last access on [2018-03-09]. [Online]. Available: <http://www.threedtk.de>
- [25] P. Falkingham. (2016) Trying all the free photogrammetry! Last access on [2018-04-26]. [Online]. Available: <https://pfalkingham.wordpress.com/2016/09/14/trying-all-the-free-photogrammetry/>
- [26] B. Horn, "Closed-form solution of absolute orientation using unit quaternions," *J. Opt. Soc. Am. A*, vol. 4, no. 4, pp. 629–642, Apr 1987.
- [27] D. Girardeau-Montaut. Cloudcompare - 3d point cloud and mesh processing software. Last access on [2018-04-06]. [Online]. Available: <http://www.cloudcompare.org/>
- [28] W. Weibull, "A statistical distribution function of wide applicability," *Journal of Applied Mechanics*, vol. 18, pp. 293–297, 1951.