TOWARDS FULLY AUTOMATIC PHOTOGRAMMETRIC RECONSTRUCTION USING DIGITAL IMAGES TAKEN FROM UAVS

A. Irschara[*,a], V. Kaufmann[b], M. Klopschitz[a], H. Bischof[a], F. Leberl[a]

[a] Institute for Computer Graphics and Vision, Graz University of Technology, Inffeldgasse 16, A-8010 Graz, Austria – {irschara,klopschitz,bischof,leberl}@icg.tugraz.at
[b]Institute of Remote Sensing and Photogrammetry, Graz University of Technology, Steyrergasse 30, A-8010 Graz, Austria – viktor.kaufmann@tugraz.at

KEY WORDS: Vision, Robotics, Reconstruction, Matching, Automation, Accuracy

ABSTRACT:

We argue that the future of remote sensing will see a diversification of sensors and sensor platforms. We argue further that remote sensing will also benefit from recent advances in computing technology to employ new algorithms previously too complex to apply. In this paper we support this argument by three demonstrations. First, we show that an unmanned aerial vehicle (UAV) equipped with digital cameras can provide valuable visual information about the Earth's surface rapidly and at low cost from nearly any viewpoint. Second, we demonstrate an end-to-end workflow to process a sizeable block of such imagery in a fully automated manner. Thirdly, we build this workflow on a novel computing system taking advantage of the invention of the Graphics Processing Unit (GPU) that is capable of performing complex algorithms in an acceptable elapsed time. The transition to diverse imaging sensors and platforms results in a requirement to deal with unordered sets of images, such as typically collected from a UAV, and to match and orientate these images automatically. Our approach is fully automated and capable of addressing large datasets in reasonable time and at low costs on a standard desktop PC. We compare our method to a semi-automatic orientation approach based on the PhotoModeler software and demonstrate superior performance in terms of automation, accuracy and processing time.

1. INTRODUCTION

Aerial photography has been the workhorse of remote sensing. Satellite imagery has augmented the remote sensing tool box since the launch of Landsat in 1972. Both aerial and satellite imaging result in very ordered and industrially planned image datasets. Recently, however, one can see a diversification of the image inputs for remote sensing (Eissenbeiss et al., 2009). Photography from handheld amateur cameras, from balloons and unmanned aerial vehicles (UAVs), all are subject to intensive research into their applicability to tasks previously reserved to industrial solutions. In the last few years, advances in material science and control engineering have turned unmanned aerial vehicles into cost efficient, flexible and rapidly deployable geodata acquisition platforms. For instance the micro-drone md4-200 (http://www.microdrones.com) depicted in Figure 1 has the ability for vertical take off and landing, provides position hold and autonomous way-point navigation and is equipped with a standard digital consumer camera that can be tilted (up to 90°) to capture images from different angles. Thus, a UAV can act as a virtual eye in the sky capable to provide visual information about an object which otherwise cannot be obtained. Therefore, photogrammetric reconstruction based on imagery taken from UAV systems is of high interest and has been addressed by many authors, e.g. in the context of digital surface model (DSM) extraction (Förstner and Steffen, 2007), archaeological preservation (Scaioni et al., 2009) and agricultural survey (Grenzdörffer et al., 2008). According to (Colomina et al., 2008), UAVs are a new paradigm for high-resolution low-cost photogrammetry and remote sensing, especially given the fact that consumer grade digital cameras provide a sufficiently high accuracy for many photogrammetric tasks (Gruen and Akca, 2008). The presence of on board navigation, Global Positioning System (GPS) and Inertial Measurement Units (IMUs) allows UAVs to act as autonomous systems that fly in the air and sense the environment. Due to the low operation altitude, UAVs achieve a very high resolution

Figure 1: Micro-drone md4-200 with attached PENTAX Optio A40.

in terms of ground sampling distance and can therefore compete with airborne large format digital camera system (e.g. UltraCamXp (http://www.microsoft.com/ultracam)).

Although, recent UAVs are most often equipped with GPS/INS positioning systems and orientation sensors, the output of these sensors does in general not achieve the required accuracy to provide direct georeferencing of the acquired imagery (Eugster and Nebiker, 2009). Hence, image based methods, referred to as structure from motion in the computer vision literature (Hartley and Zisserman, 2000), are necessary techniques to determine the exterior camera orientations. There exists a variety of approaches that address the 3D reconstruction problem from videos and ordered sets of still images, e.g. (Pollefeys et al., 2004). Real time performance for camera motion recovery on modest hardware is reported (Nistér et al., 2004), but working incrementally on a frame by frame basis leads to the inherent problem of error accumulation and drift (Steffen and Förstner, 2008). Furthermore, sequential processing is only possible for very ordered, industrially planned image datasets, such as manned airborne and spaceborne remote sensing imagery. The transition to diverse imaging sensors and platforms results in a requirement to deal with unordered sets of images. This is especially true for images captured by highly maneuverable UAV systems that allow random flight paths, hence deliver unordered image datasets.

Therefore, in practice, wide baseline matching methods that are

* Corresponding author.

able to establish geometric relations between images, which are (widely) separated in time/space, are necessary in order to obtain consistent 3D models. These methods have been shown to even work on very uncontrolled image collections such as images from the web (Snavely et al., 2006), but require a high degree of computational effort. Recently, (Agarwal et al., 2009) presented a distributed computing engine based on a cluster of 500 computing cores to automatically reconstruct 3D scenes from large image collections. Our system shares algorithmic similarities with their approach, but in contrast to rely on hundreds of computer clusters, we leverage the parallel computing power of current GPUs to accelerate several processing steps. We follow the concept of General-Purpose computing on Graphic Processing Units (GPGPU) and use Nvidia's Compute Unified Device Architecture (CUDA) toolchain for our implementation. Our proposed approach is fully automated and capable of addressing large datasets in reasonable time and at low costs on a standard desktop PC.

2. UAVS AS PHOTOGRAMMETRIC SENSOR PLATFORMS

The main advantage of a UAV system acting as a photogrammetric sensor platform over more traditional manned airborne or terrestrial surveys, is the high flexibility that allows image acquisition from unconventional viewpoints. Consider Figure 2: While the camera network in standard airborne and terrestrial surveys is normally restricted to flight lines or street paths, a UAV system enables more flexible, e.g. turntable like network configurations, that maximize scene coverage and allow superior accuracy in terms of triangulation angles. Furthermore, the photogrammetric network planning task (Chen et al., 2008) can be optimized and adapted to the scene since nearly any desired viewpoint can be reached. Moreover, networks of multiple, synchronously flying UAVs (Quaritsch et al., 2008) could be utilized to deliver multi-view information simultaneously, which opens the possibility to reconstruct also non-rigid objects over time.

The remainder of the paper is organized as follows. In the next Section we describe in detail our structure from motion system which is able to operate on unordered datasets, such as typical images captured by a UAV system. In Section 4. we show results of our method and compare our system to a standard semi-automatic approach based on the PhotoModeler software. Finally, Section 5. concludes our work.

3. 3D RECONSTRUCTION SYSTEM

Our 3D reconstruction system is able to automatically match unordered sets of images and to determine the exterior camera orientations and sparse tie points without prior knowledge of the scene. The system mainly consists of three processing steps: Feature extraction, matching and finally structure from motion computation. Figure 3 gives an overview of our reconstruction pipeline. A prerequisite of our system is that the intrinsic camera parameters are known and constant. We use the calibration method described in (Irschara et al., 2007) to simultaneously estimate the focal length, principal point and radial distortion parameters, standard values are assumed for the remaining intrinsics (i.e. zero skew and unit aspect ratio).

In general, calibrated camera settings are not strictly necessary for Euclidean 3D modeling, since self-calibration methods (Pollefeys et al., 1999) exist, but robustness and accuracy is normally greatly improved for image collections with known intrinsics. Furthermore, also an increase in processing speed is achieved due to the lower dimensionality of the problem.

http://www.nvidia.com

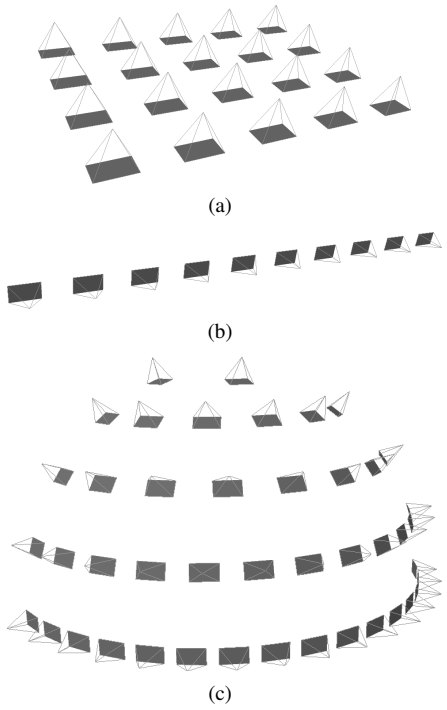


Figure 2: Typical camera networks used for aerial (a) and (b) terrestrial survey. In general, a UAV system allows the acquisition of more flexible photogrammetric camera networks, like the configuration depicted in (c), that enables a regular sampling of the visual hull of the scene of interest.

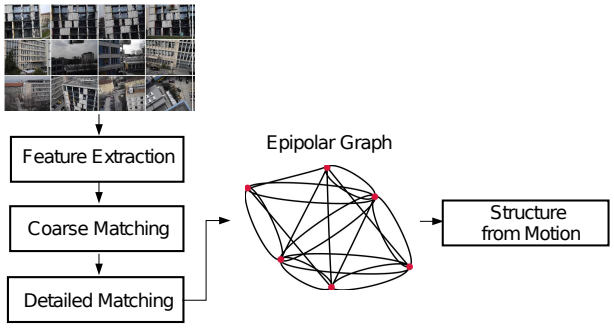


Figure 3: Overview of the main processing steps of our reconstruction pipeline.

3.1 Feature Extraction

Our system utilizes the very effective SIFT keypoint detector and descriptor (Lowe, 2004) to represent point features. SIFT features are invariant to scale and rotation and partially invariant to viewpoint and illumination changes. Hence, these kind of features are very suitable for wide baseline matching and have been found to be highly distinctive and repeatable in performance evaluation (Mikolajczyk et al., 2005). In particular we rely on the publicly available SiftGPU software. On recent GPUs, a speedup exceeding twenty over a single core CPU implementation is reached.

3.2 Matching

Unlike feature point tracking in video sequences, where correspondence search can be restricted to local regions, matching of

http://cs.unc.edu/~ccwu/siftgpu

unordered still images essentially requires exhaustive search between all image pairs and all features seen therein. Hence, the matching costs are quadratic in the total number of extracted features from the image database. Note, the number of SIFT features from a medium sized image (e.g. 4000×3000 pixel) normally exceeds a value of 10000. For a small image database consisting of 1000 images, more than 10 million SIFT keys are detected, this translates into 100 billion descriptor comparisons that are necessary for exhaustive nearest neighbor search. This is a considerable amount of computation, which turns out to be a prohibitively expensive operation executed on a single CPU.

To make the correspondence search more tractable, we divide the matching procedure into two submodules. First, we build upon work on efficient image retrieval (Nistér and Stewenius, 2006) and use a vocabulary tree to determine an image-to-image similarity score. Second, we take advantage of the high computational power of modern GPUs to establish putative correspondences between the feature sets of relevant image pairs.

3.2.1 Coarse Matching Inspired by recent advantages in image search, we use a vocabulary tree approach and inverted file voting (Sivic and Zisserman, 2003) for coarse matching of potentially similar images. The vocabulary tree based database representation is very efficient in terms of memory usage and allows an extremely fast determination (in the order of some milliseconds) whether two images are similar or dissimilar. Hence, by considering only the most relevant candidate images for pair-wise matching, the computational effort can be reduced significantly.

The vocabulary tree is constructed by offline training using hierarchical k-means clustering of millions of SIFT features (extracted from a generic image database) and gives a quantized approximation of the high dimensional descriptor space. Since k-means clustering of large datasets is a time consuming operation, we employ a CUDA based approach executed on the GPU to speed up clustering.

The vocabulary tree concept relies on the following basic assumption: if the similarity between two features $sim(f_i, f_j)$ is high, then there is a relatively high probability that the two features are assigned to the same visual word $w(f_i) \equiv w(f_j)$, i.e. the features reach the same leaf node in the vocabulary tree. Based on the quantized features from a query image $\mathcal{Q}$ and each database image $\mathcal{D}$ a scoring of relevance is derived. Typical scoring functions are based on a vector model, as for instance the *tf-idf* (term frequency, inverse document frequency), which delivers a relative document ranking according to the degree of similarity to the query. In contrast to that, in our system we rely on a scoring function that gives an absolute score of similarity based on a probabilistic model (Singhal, 2001, Irschara et al., 2009). This model allows a direct determination whether a document image is likely to match a query image.

3.2.2 Pairwise Feature Matching A variety of approaches have been proposed to speedup nearest neighbor matching in high-dimensional spaces (like the 128-dimensional SIFT descriptor space). Among the most promising methods are randomized kd-trees (Anan and Hartley, 2008) with priority search and hierarchical k-means trees (Fukunaga and Narendra, 1975). These algorithms are in general designed to run on a single CPU and are known to provide speedups of about one or two orders of magnitude over linear search, but the speedup comes with the cost of a potential loss in accuracy (Muja and Lowe, 2009). On the other hand, given that the number of features is limited to some thousands, nearest neighbor search, implemented as a dense matrix multiplication on recent graphics hardware, can achieve an equivalent speedup, but delivers the exact solution. Hence, we employ a GPU accelerated feature matching approach based on the CUBLAS library.

http://www.vis.uky.edu/ stewe/ukbench/

3.3 Epipolar Graph

After matching relevant images to each query view, geometric verification, based on the Five-Point algorithm (Nistér, 2004) is performed. Since matches that arise from descriptor comparisons are often highly contaminated by outliers, we employ a RANSAC (Fischler and Bolles, 1981) algorithm for robust estimation. In its basic implementation, RANSAC acts as hypothesize-and-verify approach. In the same spirit as (Nistér, 2005) we explicitly divide the RANSAC algorithm into two steps. First, we generate all our N relative pose hypotheses with a minimal number of five points. Second, we score the hypotheses based on the truncated Sampson error (Hartley and Zisserman, 2000) against each other. Note, the scoring procedure can be easily parallelized, hence we employ a CUDA based scoring approach in our reconstruction system.

In order to decide whether two images satisfy an epipolar geometry, we compute the RANSAC termination confidence,

$$p = 1 - \exp(N \log(1 - (1 - \epsilon)^s)) \quad (1)$$

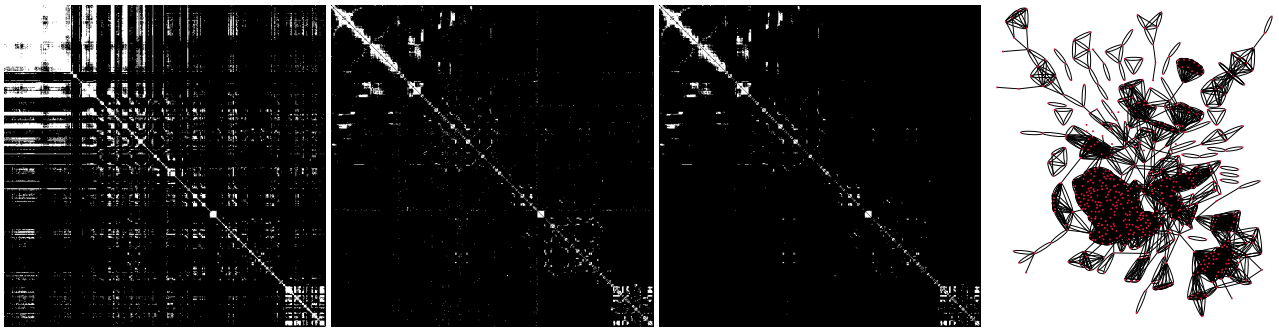
where N is the number of evaluated models, $w = 1-\epsilon$ the probability that any selected data point is an inlier, and $s = 5$ is the cardinality of the sample point set used to compute a minimal model. We require $p > 0.999$ in order to accept an epipolar geometric relation. In our experiments, we used up to $N = 2000$ models which corresponds to a maximal outlier fraction of $\epsilon = 0.67$. The epipolar graph of the UAV-dataset is shown in Figure 4(d).

3.4 Structure from Motion

Our structure from motion approach follows a greedy strategy, similar to the one described in (Irschara et al., 2007). Starting from a reliable image triplet, new views are incrementally registered by robust camera resectioning based on the Three-Point algorithm (Haralick et al., 1991) inside a RANSAC loop. Incremental Euclidean bundle adjustment is used to simultaneously refine structure (3D points) and motion (camera matrices).

4. RESULTS AND DISCUSSION

In our experiments we performed two test-flights with the micro-drone md4-200 and an attached PENTAX Optio A40 camera as depicted in Figure 1. The camera was precalibrated and the zoom was fixed to a wide angle setting. The survey was performed by manual remote control, 615 still images with a resolution of 4000×3000 square pixels were captured from different viewpoints. Furthermore, eight ground control points were determined using a total station (with an accuracy of $\epsilon \pm 1cm$, see Figure 8). This data is considered as ground truth and is later used to asses the object space error of the automatic computed structure from motion results. Figure 4 shows the affinity matrix according to the probabilistic scoring used for coarse matching. On average each image is only matched with 84 potentially similar views, which gives a speedup of approximately seven compared to a full exhaustive search. Still, 86% of potential epipolar relations are found. Note, the average degree of image overlap in this dataset is relatively high. A much higher speedup would be achieved if one considers larger datasets with a sparser image overlap. Since the epipolar graph of the UAV-datasets is not fully connected (see Figure 4(d)), several individual 3D reconstructions are obtained.



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

Figure 4: (a) Image affinity matrix according to the probabilistic model of relevance. (b) Epipolar adjacency matrix computed by exhaustive image matching and geometric verification. A white entry in the matrix indicates that an epipolar geometry between two images I_i and I_j could be computed. (c) Successfully recovered epipolar geometries by considering only relevant images according to (a). (d) Epipolar connectivity graph of the whole dataset, clusters in the graph represent a high degree of geometric connectivity.

	CPU [s]	GPU [s]
SIFT (4000×3000 pixel)	10	0.4
Coarse Matching	0.5	0.05
Matching (5000×5000)	$k \times 1.1$	$k \times 0.044$
RANSAC-H (5-pt, N=2000)	$k \times 0.1$	-
RANSAC-V ($\|C\|$=5000, N=2000)	$k \times 0.12$	$k \times 0.02$
Structure from Motion [h]	1	-
Total Time [h] (615 views, $k = 84$)	21	3.5

Table 5: Comparison of processing timings between execution on a single core CPU (Intel Pentium D 3.2Ghz) vs. a GPU accelerated implementation (Nvidia GeForce GTX280). RANSAC-H stands for the hypotheses generation step based on the Five Point algorithm, RANSAC-V for the evaluation module. N is the maximal number of hypothesis, $|C|$ the number of putative correspondences used for evaluation, and k reflects the number of considered images for detailed feature matching and geometric verification.

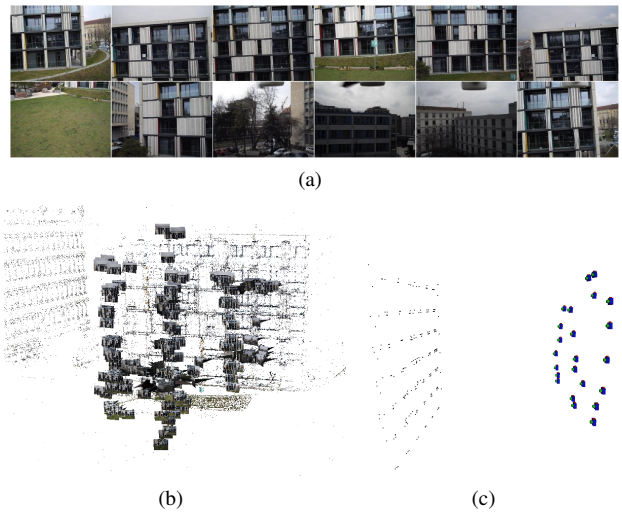
Figures 6 and 9 show visual results of the two largest connected reconstruction results, denoted as R1 (239 registered images) and R2 (68 registered images) through our experimental evaluation.

Table 5 gives typical processing times of the modules involved in our system and compares timings of a single CPU execution with timings achieved with GPGPU support. Regarding feature extraction and matching, the speedup induced by the GPU is about one order of magnitude.

4.1 Accuracy Analysis

We compare our fully automatic structure from motion approach to the semi-automatic PhotoModeler software (version 6) for the task of exterior image orientation. Since it turns out that processing 615 images is impracticable for a semi-automatic system, we restrict our evaluation to a subset of 23 manually selected images from one building facade (corresponding to result R1, see Figure 6). The processing steps of the PhotoModeler approach include the semi-automatic measurement of tie and control points, bundle adjustment and fine tuning. Four different orientation methods were conducted: selfcalibration with constant/variable intrinsics and with/without reference point constraints by using fifteen 3D control points, respectively. All methods give consistent results, on average a reprojection error of 0.5 pixel is reported. A detailed, quantitative comparison of the PhotoModeler

http://www.photomodeler.com



(a) (b) (c)

Figure 6: Orientation result R1: (a) Sample input images and (b) perspective view of camera orientations (239) and respective 3D points (58791) obtained by our automatic structure from motion system. (c) Orientation result obtained by semi-automatic processing using the PhotoModeler software, a subset of 23 manually selected images is used.

orientation output with results from our structure from motion pipeline is summarized in Table 7.

The semi-automatic approach, based on the PhotoModeler software, was performed by an expert user, the orientation of a subset of 23 images still requires about eight man hours (and is troublesome and strenuous work). On the other hand, with our fully automated system, all 615 images can be processed at once and within a timeframe of 3.5 hours on a standard PC and a single GPU. We achieve identical results in terms of reprojection error, but with a higher confidence in the solution, since many more tie points are utilized. Furthermore, the automatic approach is scalable and allows registration of many more images much faster. For instance, in our pipeline processing one image takes about 20s, whereas orientation with the PhotoModeler software requires more than 20min man workload.

4.1.1 Object Space Error The reprojection error is a suitable measure to assess the precision of camera orientations in image space, but for a practical application, the error in object space is of interest. Therefore, we rely on control points measured by a

	PhotoModeler	sfm-approach
# processed views	**23**	**615**
# registered views (R1)	23	239
# 3D points	237	58791
avg. # points/image	**99**	**3160**
avg. # rays/3D point	10	13
avg. triangulation angle	10°	6.7°
avg. reprojection error	0.458	0.460
processing time [h]	8	3.5
processing time/image [s]	**1252**	**20**

Table 7: Comparison of the semi-automatic PhotoModeler orientation to our proposed fully-automatic structure from motion system (sfm-approach), the values correspond to reconstruction result R1 (see Figure 6).

total stations to estimate an absolute error measure. The landmarks are determined at well localized structures, like building corners and junctions (see Figure 8). Thus, image measurements with respect to the corresponding landmark are easily to establish. For each image we estimate the 2D coordinates belonging to the 3D control point (manually by visual inspection) and link the measurements into point tracks. In practice, we only use a subset of images to measure observations, but ensure that for each control point at least three measurements are provided and the triangulation angle is sufficiently high ($\bar{\alpha} > 20°$). Next, we use a linear triangulation method (Hartley and Zisserman, 2000) followed by bundle-adjustment to triangulate the measurements into 3D space. In order to measure the object space error, we compute the 3D similarity transform between 3D control points and respective triangulated tie points. The alignment can be computed with a minimal number of three point correspondences, but using more than three points in a least squares manner will result in a closer alignment. Hence, we use the leave-one-out cross-validation (Kohavi, 1995) technique to assess the accuracy of our orientation results. We take seven correspondences to compute the parameters for the similarity transform and use the remaining point to estimate the object space error ϵ between observation X and ground truth point $\hat{X}$,

$$\epsilon = \sqrt{(X_x - \hat{X}_x)^2 + (X_y - \hat{X}_y)^2 + (X_z - \hat{X}_z)^2}. \quad (2)$$

Table 10 summarizes our evaluation, the error varies between 0.4 to 5.4cm, overall a RMSE of 3.2cm is achieved. Note, the reprojection error of the triangulated tie points varies between $1.1-2.5$ pixel, this is in accordance to the expected uncertainty induced by the manual tie point extraction. A subpixel accurate measurement of tie points (e.g. 0.5 pixel) would lead to a RMSE of about 1.5cm, that is close to the precision of the total station.

5. CONCLUSIONS

In this paper we demonstrated the feasibility of accurate and fast 3D scene reconstruction from unordered images captured by a UAV platform. We compared the orientation results of our fully automatic structure from motion pipeline to a standard, semi-automatic approach based on the PhotoModeler software. From our experiments we conclude that our system achieves the same accuracy in terms of reprojection error, but at a higher confidence, since many more tie points are utilized than for the semi-automatic approach. Furthermore, our method is scalable to larger datasets and allows much faster image orientation. In our experiments we achieve a speedup of about 60 over semi-automatic processing with the PhotoModeler software.

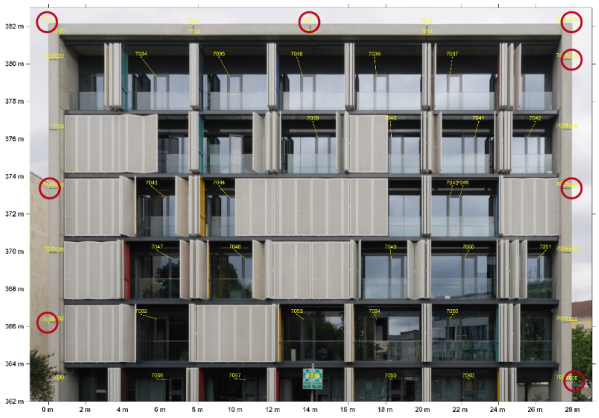


Figure 8: Orthographic projection of a building facade with the eight ground truth control points (red circles) used in our evaluation.

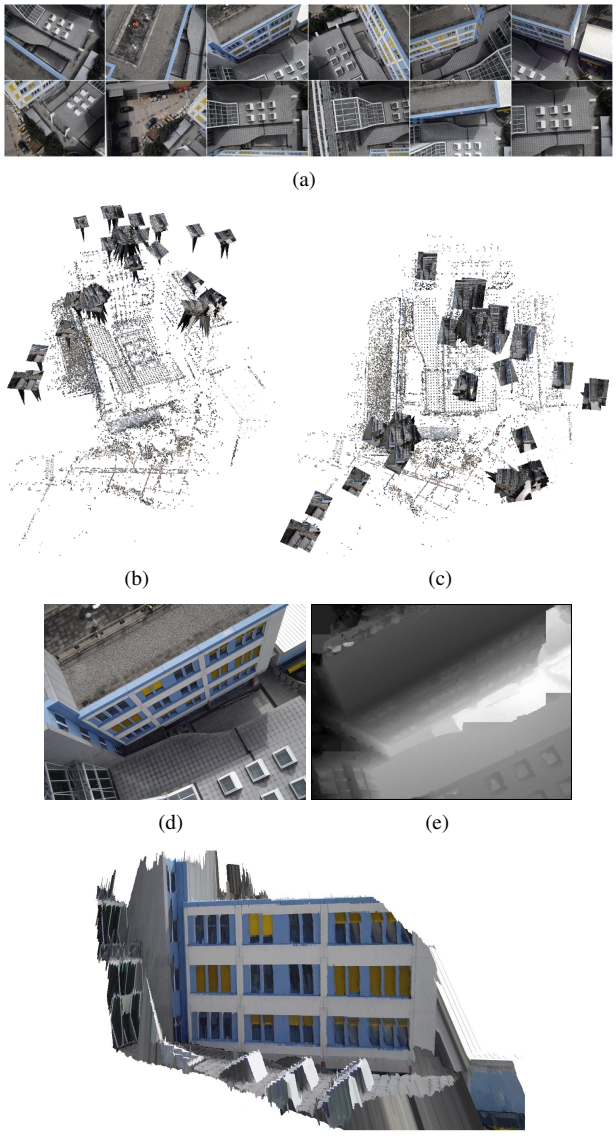


Figure 9: (a) Sample input images and (b),(c) perspective view of camera orientations and respective 3D points.(e) Input image and related depth map (f) obtained by dense matching techniques. (f) Texturized depthmap from an oblique viewpoint.

Point ID	7000	7006	7010	7012	7021	7017	7025	7029
# measurements (images)	3	6	3	3	10	3	10	6
avg. triangulation angle [°]	107.2	21.9	23.2	23.2	33.4	54.7	69.5	84.6
avg. reprojection error [pixel]	1.18	1.67	2.24	1.63	1.58	1.16	2.44	0.85
object space error [cm]	4.2	0.4	2.5	4.5	0.6	2.8	1.7	5.4

Table 10: Reprojection error and object space error determined by leave-one-out cross-validation for eight ground truth control points.

REFERENCES

Agarwal, S., Snavely, N., Simon, I., Seitz, S. M. and Szeliski, R., 2009. Building Rome in a day. In: IEEE International Conference on Computer Vision (ICCV).

Anan, C. S. and Hartley, R. I., 2008. Optimised KD-trees for fast image descriptor matching. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1–8.

Chen, S. Y., Li, Y. F., Zhang, J. W. and Wang, W. L., 2008. Active Sensor Planning for Multiview Vision Tasks. Springer-Verlag.

Colomina, I., Blázquez, M., Molina, P., Parés, M. and Wis, M., 2008. Towards a new paradigm for high-resolution low-cost photogrammetry and remote sensing. In: The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. XXXVII, Part B1, pp. 1201–1206.

Eissenbeiss, H., Nackaerts, K. and Everaerts, J., 2009. UAS for mapping & monitoring applications. In: 2009/2010 UAS Yearbook - UAS: The Global Perspective, 7th Edition, pp. 146–150.

Eugster, H. and Nebiker, S., 2009. Real-time georegistration of video streams from mini or micro UAS using digital 3D city models. In: 6th International Symposium on Mobile Mapping Technology.

Fischler, M. A. and Bolles, R. C., 1981. Random sample consensus: a paradigm for model fitting with application to image analysis and automated cartography. Communication Association and Computing Machine 24(6), pp. 381–395.

Förstner, W. and Steffen, R., 2007. Online geocoding and evaluation of large scale imagery without GPS. In: D. Fritsch (ed.), Photogrammetric Week '07, Heidelberg, pp. 243–253.

Fukunaga, K. and Narendra, P. M., 1975. A branch and bound algorithm for computing k-nearest neighbors. IEEE Transactions on Computers C-24(7), pp. 750–753.

Grenzdörffer, G., Engel, A. and Teichert, B., 2008. The photogrammetric potential of low-cost uavs in forestry and agriculture. In: The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. XXXVII, Part B1, pp. 1207–1214.

Gruen, A. and Akca, D., 2008. Metric accuracy testing with mobile phone cameras. In: The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. XXXVII, Part B5, pp. 729–736.

Haralick, R. M., Lee, C., Ottenberg, K. and Nölle, M., 1991. Analysis and solutions of the three point perspective pose estimation problem. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 592–598.

Hartley, R. and Zisserman, A., 2000. Multiple View Geometry in Computer Vision. Cambridge University Press.

Irschara, A., Zach, C. and Bischof, H., 2007. Towards wiki-based dense city modeling. In: Workshop on Virtual Representations and Modeling of Large-scale environments (VRML).

Irschara, A., Zach, C., Frahm, J. M. and Bischof, H., 2009. From structure-from-motion point clouds to fast location recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2599–2606.

Kohavi, R., 1995. A study of cross-validation and bootstrap for accuracy estimation and model selection. In: IJCAI, pp. 1137–1145.

Lowe, D., 2004. Distinctive image features from scale-invariant keypoints. Int. Journal of Computer Vision 60(2), pp. 91–110.

Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T. and Van Gool, L., 2005. A comparison of affine region detectors. Int. Journal of Computer Vision 65, pp. 43–72.

Muja, M. and Lowe, D. G., 2009. Fast approximate nearest neighbors with automatic algorithm configuration. In: A. Ranchordas and H. Araújo (eds), VISAPP (1), INSTICC Press, pp. 331–340.

Nistér, D., 2004. An efficient solution to the five-point relative pose problem. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) 26(6), pp. 756–770.

Nistér, D., 2005. Preemptive RANSAC for live structure and motion estimation. Mach. Vis. App.

Nistér, D. and Stewenius, H., 2006. Scalable recognition with a vocabulary tree. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2161–2168.

Nistér, D., Naroditsky, O. and Bergen, J., 2004. Visual odometry. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 652–659.

Pollefeys, M., Koch, R. and Gool, L. V., 1999. Self-calibration and metric reconstruction in spite of varying and unknown internal camera parameters. Int. Journal of Computer Vision 32(1), pp. 7–25.

Pollefeys, M., Van Gool, L., Vergauwen, M., Verbiest, F., Cornelis, K., Tops, J. and Koch, R., 2004. Visual modeling with a hand-held camera. Int. Journal of Computer Vision 59(3), pp. 207–232.

Quaritsch, M., Stojanovski, E., Bettstetter, C., Friedrich, G., Hellwagner, H., Rinner, B., Hofbaur, M. W. and Shah, M., 2008. Collaborative microdrones: applications and research challenges. In: A. Manzalini (ed.), Autonomics, p. 38.

Scaioni, M., Barazzetti, L., Brumana, R., Cuca, B., Fassi, F. and Prandi, F., 2009. RC-heli and structure and motion techniques for the 3-D reconstruction of a milan dome spire. In: 3DARCH09.

Singhal, A., 2001. Modern information retrieval: A brief overview. IEEE Data Engineering Bulletin 24(4), pp. 35–43.

Sivic, J. and Zisserman, A., 2003. Video google: A text retrieval approach to object matching in videos. In: IEEE International Conference on Computer Vision (ICCV), pp. 1470–1477.

Snavely, N., Seitz, S. and Szeliski, R., 2006. Photo tourism: Exploring photo collections in 3D. In: Proceedings of SIGGRAPH 2006, pp. 835–846.

Steffen, R. and Förstner, W., 2008. On visual real time mapping for unmanned aerial vehicles. In: The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. XXXVII, Part B3a, pp. 57–62.

ACKNOWLEDGEMENTS

We are grateful to Wolfgang Waagner for providing and piloting the micro-drone md4-200. This work was supported by the Austrian Science Fund (FWF) under the doctoral program Confluence of Vision and Graphics W1209 and the PEGASUS (825841) project, financed by the Austrian Research Promotion Agency (FFG).