Online Feedback for Structure-from-Motion Image Acquisition

Christof Hoppe[1]
hoppe@icg.tugraz.at
Manfred Klopschitz[1]
klopschitz@icg.tugraz.at
Markus Rumpler[1]
rumpler@icg.tugraz.at
Andreas Wendel[1]
wendel@icg.tugraz.at
Stefan Kluckner[2]
stefan.kluckner@siemens.com
Horst Bischof[1]
bischof@icg.tugraz.at
Gerhard Reitmayr[1]
reitmayr@icg.tugraz.at

[1] Institute for Computer Vision and Graphics
Graz University of Technology
Graz, Austria

[2] Research Group Video Analytics
Corporate Technology
Siemens AG Austria, Graz

Abstract

The quality and completeness of 3D models obtained by Structure-from-Motion (SfM) heavily depend on the image acquisition process. If the user gets feedback about the reconstruction quality already during the acquisition, he can optimize this process. We propose an online SfM approach that allows the inspection of the current reconstruction result on site. To guide the user throughout the acquisition, we visualize the current Ground Sampling Distance (GSD) and image redundancy as quality indicators on the surface model. The contributions of this paper are an online SfM framework for high-resolution still images that achieves an accuracy close to an off-line SfM method and a visualization of quality measures that allow the user to optimize the image acquisition process. We compare the accuracy of the proposed online SfM to state-of-the-art batch-based SfM methods and demonstrate how our algorithm improves the acquisition process.

1 Introduction

Structure-from-Motion became very popular in the last years for the reconstruction of large-scale environments. The increasing computational power and the increasing number of publicly available SfM pipelines like Bundler [22] leverages SfM for a large number of applications. Application areas vary from documentation of time-changing environments such as growing construction sites [6] to manned image collection projects like aerial surveys. Since

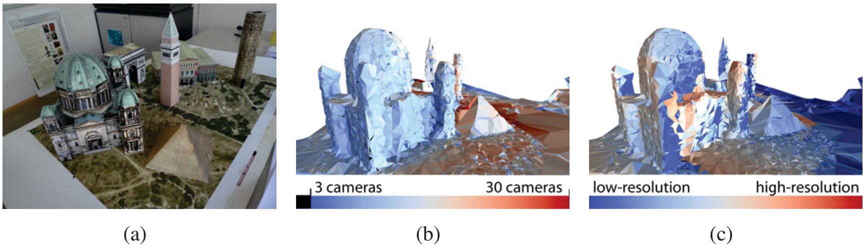


Figure 1: (a) Sample image of the City-of-Sights dataset. (b) Final mesh overlaid by redundancy information. (c) The resolution visualization, e.g. the GSD supports the user to recognize which parts of the scene have been sampled at which resolution. Best viewed in color.

most of them require very detailed 3D reconstructions, they are based on high-resolution still images.

The quality and completeness of a reconstructed model largely depends on the image quality and on its acquisition strategy. Hence, for a predictable application of SfM it is important to guarantee that the images taken on site are sufficient for the reconstruction. For example when documenting a construction site it is not possible to repeat image acquisition because the scene changes over time rapidly. In other scenarios a retake is possible but might be very expensive, for example, when images are acquired from aerial viewpoints. Furthermore, in many applications the goal is to obtain a dense reconstruction which is computationally demanding [5, 9, 17]. Therefore, it is important to know that the acquired images are suitable to achieve the expected quality before performing the dense reconstruction.

To recover a detailed and complete 3D model, the SfM process has several requirements on the input images: The viewing angle between two images may not be too large to allow feature matching, the view cones must overlap, the images have to be textured but the texture may not be repetitive and lighting hasn't changed too much between images. For a user it is impossible to estimate if the acquired images fulfill all demands. Another difficult question is the scene completeness, i.e. the coverage of the scene. Parts of the scene that are not captured with sufficient image overlap cannot be reconstructed. Since completeness depends on the required reconstruction resolution i.e. level-of-detail, and on the surface itself it is not possible to quantify the degree of completeness without prior information. In contrast, for a human it is relatively easy to answer this question by comparing a 3D model to the real world.

The problem of calculating view points for scene exploration is also known as Next-Best-View (NBV) problem in robotics. Here, mostly the scene is reconstructed online using a video stream [24] or, if computational power is not available, prior information is employed [10]. Our goal is to support a human who manually acquires still images and who has no prior information of the scene. Therefore, our method is designed as an interactive human-in-the-loop system where the user locates the view points and judges if reconstruction quality is sufficient.

The main question throughout this paper is: How can we make the SfM results predictable during image acquisition? Therefore we address the following problems:

- How to obtain accurate and fast SfM results during image acquisition.
- How to support the user to achieve complete reconstructions.

Our proposed solution is an online SfM algorithm that delivers accurate 3D information comparable to off-line state-of-the-art SfM approaches with a short response time. We obtain a surface model from the sparse point cloud and visualize quality information that helps to predict the final SfM result. Figure 2 illustrates the workflow of our method. Our approach can be used in varying scenarios from images obtained by a human user to images obtained by a mobile image acquisition platform like micro aerial vehicles (MAVs).

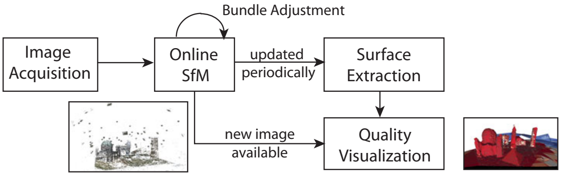


Figure 2: Workflow. Each time a new image is acquired, it is integrated into the map and if a mesh has already been created, the GSD and the degree of redundancy is updated immediately. In a parallel thread, the map is optimized by bundle adjustment. The mesh is created periodically and also runs in a parallel thread.

2 Related Work

Our feedback system is based on an online SfM algorithm that produces very accurate camera poses as well as a precise 3D structure. It is designed for users that manually take high-resolution still images with a consumer camera with non-constant framerates. We cannot employ Simultaneous Localization and Mapping (SLAM) algorithms as proposed for example by Klein and Murray [13] to obtain camera poses online. Such systems require constant and high framerates for camera tracking. Furthermore, they can select an arbitrary image of the videostream to extend the map whereas we have to handle each image that is provided by the user.

Hence, our framework is closer to off-line SfM methods [11, 22]. These approaches assume that all images are available at the programs start. They exhaustively match all images against each other and construct an epipolar graph to obtain spatial ordering. Using this graph, they incrementally reconstruct camera positions and 3D structure. To obtain a consistent map, bundle adjustment is performed after each expansion step. In contrast to these approaches, we do not require an epipolar graph of all images and therefore we can integrate images in an online fashion.

Recently, several approaches have been presented that derive the surface of an object in real-time from a video stream [7, 18]. Given a set of images, they perform planesweep to generate dense depth maps and fuse them using variational methods. Since this is typically performed on the GPU, the voxel resolution is limited to the GPU memory. Furthermore, the volume has to be initialized by setting the bounding box of the reconstruction and cannot be changed afterwards. In contrast, Pan *et al*. proposed a system called ProFORMA [21] to rapidly produce 3D models through a 3D Delaunay triangulation [2] of the sparse point cloud, followed by a probabilistic carving step to remove those triangles on the outside of the object, violating the visibility constraint. The robustness to outlier points is limited to small Gaussian noise. Labatut *et al*. [15] use a similar idea to extract a surface mesh. They also perform a 3D Delaunay tetrahedralization and define an energy functional that is minimized to label thetraheda as in- and outside. The optimization is performed by graph cuts [3]. The result is a watertight surface mesh that is free of self intersections and insensitive to outliers. An advantage over volumetric methods is that the level-of-detail problem is solved inherently. Parts of the surface that are sampled more densely by 3D points are represented with a higher resolution than weakly sampled parts, whereas the voxel representation has a uniform resolution. This property allows to extract the surface from a non-uniformly sampled surface.

3 Online SfM for Wide-baseline Still-images

Classically, batch-based SfM approaches assume spatially unordered images as input and therefore require several minutes or hours to determine the spatial ordering by constructing an epipolar graph [22]. To provide direct feedback of the current reconstruction quality to a user, we have to solve the SfM problem in an online fashion. SLAM systems like [13] are able to recover camera positions and 3D structure in real-time, but they rely on high framerates and strong motion assumptions and therefore they are not suitable for our purpose.

Since the construction of the epipolar graph comprises the calculation of relative orientations between all image pairs, this is the most time-consuming task in batch SfM pipelines. In our addressed problem, we can assume that a user does not acquire images in a totally random order. If we assume that a new input image I has an overlap to an already reconstructed scene part, we can skip the epipolar graph construction and the SfM problem can be split into two tasks that are easier to solve: A localization and a structure expansion part. More formally, given a freshly acquired input image I and a reconstructed scene M, we find the position of I within M and finally, we expand the map M. For bootstrapping the scene M, we rely on initialization schemes that are also used in batch-based SfM methods.

To calculate the pose of an image with respect to an SfM point cloud we follow the approach of Irschara *et al.* [12]. Given a new image I, we compare its visual appearance against all already reconstructed images. This can be efficiently computed by a vocabulary tree as proposed by Nister and Stewenius [20] and results in a similarity score for each image. Then we match I pairwise against the image features of the top n images with highest similarity score to determine feature correspondences. Since some of the features are already used for the triangulation of 3D points, we can establish 2D-3D image correspondences between I and M. Given a set of 2D-3D correspondences and a calibrated camera, we solve the absolute pose problem [14] robustly in a RANSAC [4] loop. If a valid position for I is determined, this pose is refined by minimizing the reprojection error using non-linear optimization [1]. Since we even have to cope with large baselines between images, we use SIFT features [16] for feature matching.

If we cannot localize I within M, this is reported to the user instantly. Hence, he directly knows that I could not be aligned within the map and he is asked to take a new picture.

Since the orientation of I is known and we have already image correspondences available from the previous step, we can easily triangulate new 3D points. To reduce the number of outlier matches, we perform epipolar filtering before triangulation.

For bootstrapping the initial map M, we require two images taken from different viewpoints, and perform brute-force feature matching. The Five-Point pose estimation algorithm of Nister [19] in a RANSAC loop is used to find the relative orientation between both cameras. Using the inlier correspondences returned by RANSAC, we triangulate 3D points.

To prevent the triangulation of degenerated 3D points lying at infinity, we require that the triangulation angle exceeds a minimum threshold of 2 degrees. We also require that the initial map consists of a minimum number of triangulated points, e.g. 100. If this cannot be achieved, the user is asked to take new images until the system finds an image pair that fulfills the requirements.

To prevent scene drift caused by incremental map building, we use a global optimization scheme to obtain a consistent map. Hence, we perform iterative bundle adjustment [23] in a parallel thread. For feature extraction and matching we employ the GPU to meet the real-time requirements.

4 Scene Completeness and User Interaction

Without feedback it is even for a very experienced user difficult to determine if a large-scale scene is sampled sufficiently, i.e. if all relevant parts have been captured. Although the SfM point cloud already indicates the scene completeness, it is difficult for a user to derive this information from the point cloud. Therefore, we extract a triangular surface mesh from the SfM point cloud. Based on the mesh, we calculate two quality measures (GSD and the degree of redundancy) that support the user in predicting the final dense reconstruction result. The GSD measures the resolution of the mapping between the 3D physical surface to 2D image space. Therefore, the GSD determines the (theoretical) maximum resolution of a dense 3D model. The degree of redundancy describes how often a physical surface area is captured by images. Redundancy is required to achieve accurate results but it also increases the computational costs and lengthens the acquisition. Especially in large-scale and geometrically complex scenes, the visualization of the redundancy supports the user to obtain complete scene sampling.

4.1 Surface Extraction

We employ the approach of Labatut *et al.* [15] to extract a surface mesh given the SfM point cloud. The idea is to generate a 3D triangulation of all sparse 3D points that embeds the real surface. To extract the subset of triangles which are on the object's surface, an energy functional based on visibility information is defined, and minimized by graph cuts. Compared to the original formulation, we neglect the photo consistency part of the energy functional for computational reasons. The result of the algorithm is a watertight triangular mesh. In contrast to a volumetric method, this approach is not limited to a bounding volume and is able to extract a surface even from a very sparse point cloud.

4.2 Quality Measures

Based on the extracted surface model, we derive two measures to support the user's acquisition process: The GSD and the degree of redundancy. To evaluate the GSD, we calculate the maximum resolution a mesh triangle is mapped to in image space. We reproject each triangle T_i of the mesh S to each aligned camera C_t. We then calculate the maximum resolution which corresponds to the minimum value of the GSD

$$R(T_i) = \min_{C_t} \sqrt{\frac{A(T_i)}{P(T_i, C_t)}} \qquad (1)$$

where $P(\cdot,\cdot)$ is the number of pixels that triangle T_i covers in camera C_t and $A(\cdot)$ is the area of the triangle in 3D space. To handle self-occlusions of the mesh correctly and for efficient calculation we employ the GPU that is optimized for visibility estimation of meshes. For calculating $R(T_i)$, we assign a unique color to each triangle. We then render S using OpenGL from viewpoint C_t and read out the image buffer. We calculate $P(T_i, C_t)$ by counting the pixels that have the color assigned to T_i.

The degree of redundancy can be computed at the same time by counting the number of of cameras T_i is visible in. We define that T_i is visible in C_t if less than 50% of T_i's area is occluded. This prevents triangles from being counted as visible that are largely occluded. Since the mesh is rendered on graphics hardware and only counting is performed on the CPU, the coverage computation takes around 50 ms for a single viewpoint.

To visualize both measures, we overlay the mesh by a color map according to the measure's value. The user interactively selects which information he requires for the decision on his next step. Since the scale of the SfM result is arbitrary, $A(\cdot)$ is typically not in metric scale and we determine the range of the color map by α-trimming all values of $R(T_i)$ where $\alpha = 10\%$. If the scale of the reconstruction can be determined, for example by aligning the reconstruction to GPS data, we can choose the color map according to predefined resolutions. Figure 1(c) demonstrates this visualization for the reconstructed City-of-Sights [8] paper model.

4.3 User Interaction

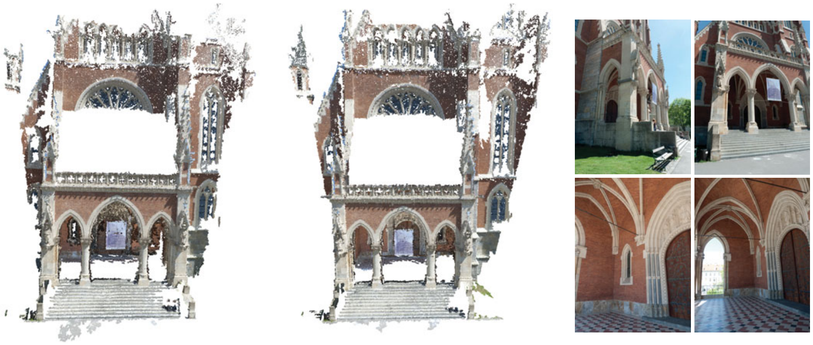
Each time the user acquires a new image, we try to register it to the SfM point cloud, and on success we triangulate new 3D points as described in Section 3. If the localization of an image fails, this is reported to the user and he can adopt his acquisition strategy. After a few images are integrated into the map, we extract a surface model and calculate the GSD and the degree of redundancy. Each time a new image is inserted into the map both quality measures are updated for each triangle and provided to the user. Since surface extraction takes some seconds to finish, the mesh structure is updated according to available computational power. The workflow is illustrated in Figure 2 and a demonstration video can be found on http://www.aerial.icg.tugraz.at/.

5 Experiments

In our experiments, we show that the results of the online SfM are similar to the results obtained by a batch-based state-of-the-art SfM regarding accuracy while being about 7 times faster. We also show that our method helps even a very experienced user to increase the number of images that are suited for a reconstruction. To demonstrate the versatile application areas our approach can be used in, we show results based on outdoor images acquired by a flying octo-rotor helicopter and the reconstruction of a small paper model indoors. All images are acquired by a Panasonic DMC LX3 (10 Mpx) equipped with a Toshiba Flashair WIFI card for live image transmission.

5.1 Online SfM vs. Batch-based SfM

To compare the accuracy of online SfM to a state-of-the-art batch-based SfM approach, we acquired an image sequence of 74 outdoor images of a church entry (Figure 3) and recon-



(a) (b) (c)

Figure 3: Comparison of dense results obtained by PMVS [5] when using the sparse result calculate by our online SfM (a) and the result achieved by Bundler (b). Both are visually similar and also the number of reconstructed 3D points is comparable. (c) Example of input images.

	SfM Runtime	# sparse	avg measurements	# dense
		points	per 3D point	points
Bundler	930 s	28,215	3.33	2,416,144
our online SfM	129 s	23,218	3.85	2,399,999

Table 1: Comparison Bundler vs. our online SfM

structed them using both SfM methods. Because it is difficult to generate ground truth data for a large-scale outdoor dataset, a good indicator is the result of the dense matching step. Small errors in the camera alignment have large effects in the final dense reconstruction.

For both methods, we use 4000 SIFT features per image that have largest scale. The features are extracted by the SIFTGPU [25] implementation. Table 1 shows the comparison between both methods. Our approach requires 129 seconds which is 7.2 times faster than Bundler. Our approach generates less 3D points which is because we use only $n = 6$ images for the triangulation of new points. In contrast, we obtain an increased number of measurements per 3D point compared to Bundler. This allows to conclude that our cameras are connected more densely and therefore we can expect a similar accuracy. On average, our approach requires 1.8 seconds to integrate a new image into the map and to extend the map. Since we fix the number of images for matching, the insertion time is independent of the map size (apart from bundle adjustment which runs in a parallel thread). The timings for

	Time
Undistortion	280 ms
SIFT feature extraction (GPU)	730 ms
Vocabulary tree scoring	40 ms
Feature Matching (GPU)	600 ms
Localization absolute pose	20 ms
Triangulation of new points	20 ms

Table 2: Timings for the integration of a 10 Mpx image into the reconstruction. Timings are determined on a 3.2GHz Intel Core i7 processor and an NVIDIA GTX 580 graphics card.

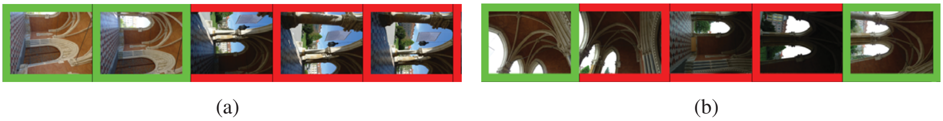


Figure 4: Sequence of acquired images. (a) If acquiring images without online feedback, the user did not recognize that images cannot be aligned to the reconstruction (red). The next 20 images could not be added incrementally. (b) With feedback, the user recognized the problem and reacted on that. Therefore, the following images are inserted correctly.

each step are given in Table 2. The undistortion and feature extraction are dependent on the image size whereas the subsequent feature matching and map extension steps only depend on the number of features used for matching. Hence, these two parameters can be adjusted if less computational power is available or if faster image integration is needed.

Figure 3 shows the densified SfM point cloud when using the sparse SfM result obtained by our online approach and the result when using the off-line reconstruction of Bundler. Both point clouds have nearly the same number of 3D points and their visual appearance is also very similar. This demonstrates that the online SfM result is accurate and can be used directly for subsequent processing steps like dense matching.

5.2 User Support

To demonstrate that our approach supports the user during image acquisition, we performed an experiment with a user that has deep knowledge about SfM methods and is familiar with image acquisition for 3D reconstruction. We asked the user to acquire images of a church entrance that are processed by our online SfM algorithm. We advised him to take images that have enough overlap to be integrated into the existing reconstruction. We were interested on an outside reconstruction as well as on the reconstruction of the vaulted ceiling, which made image acquisition more complicated because of the non-convex object's shape (see Figure 4). We performed the experiment twice: Without feedback and with online feedback. During the first experiment without feedback he was advised to acquire 100 images, in the second experiment with feedback he should stop image acquisition once 100 images were integrated into the reconstruction by our system.

Without feedback, our method successfully integrated 74 of 100 images into a consistent reconstruction. Most images that cannot be reconstructed were acquired when looking from the inside of the entrance to the brighter background. The dynamic range of the camera is insufficient and parts of the vaulted ceiling are underexposed. Since the user did not recognize this, a sequence of 20 images are missing in the reconstruction. When incorporating feedback, the user recognized this problem after 3 images because our algorithm reports that the underexposed images could not be integrated into the reconstruction and the user adjusted the exposure settings. Figure 4 illustrates the difference between both experiments. He captured 118 images to achieve that 100 images are integrated into the reconstruction,which is a rate of 15% missed images compared to 26% in the experiment without feedback.

5.3 Application Areas

We performed an experiment on a large outdoor sceen as well as on a small paper model to demonstrate the variety of scenes our approach can cope with. The images of an atrium

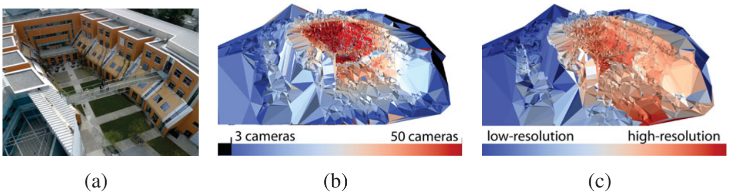


Figure 5: Results of the Atrium scene. (a) Sample image from the scene acquired by an MAV. (b) Redundancy map. (c) Visualization of the GSD. Best viewed in color.

are acquired by a manually controlled MAV at very different heights. The paper model is captured manually by a user. Figure 5 shows a sample image of the atrium dataset and the final mesh overlaid by the redundancy and resolution information. The overall surface is extracted correctly even from a very low number of 3D points. The redundancy map shows that the user mostly concentrated on the back part of the building. The resolution map shows that the resolution is distributed equally within the atrium. The reconstruction of the paper model is shown in Figure 1. Figure 6 demonstrates the evolution of the mesh over time. With increasing number of 3D points, the mesh gets more detailed but already after 10 images, the rough shape of the object is observable and the color coding helps the user to select new camera positions. Table 3 summerizes significant results of the final reconstruction. The table gives evidence that the time is constant for image integration and the meshing times are sufficient to provide online feedback.

6 Conclusion and Future Work

We have presented an approach that supports users during image acquisition for complete SfM results in three different ways: We calculate the SfM result online and provide feedback if the last acquired image can be aligned within the reconstruction. Furthermore, we extract a triangular surface mesh which allows the user to judge if all parts of the scene are reconstructed, and finally, we calculate the GSD and the degree of redundancy to guide the next acquisition steps of the user.

We have shown that our online SfM integrates new high-resolution images in less than 2 seconds and achieves an accuracy that is very similar to off-line methods such as Bundler. The result can be directly fed in a dense reconstruction pipeline like PMVS and makes a time consuming post-processing SfM needless. The user experiment has shown that our method reduces the number of acquired images that are not sufficient for SfM.

In the future, we will extend our method to provide suggestions for good view points. Even without this feature, our method supports users to judge the quality and the completeness during image acquisition on site and makes the SfM result predictable.

	Images	3D points	SfM time	Triangles	Meshing time
City-of-Sights	61	22,752	110 s	18,810	21 s
Atrium	127	28,374	238 s	22,798	19 s

Table 3: Reconstruction results for the two different scenes. The meshing time includes the time needed for calculating the GSD and the redundancy information.

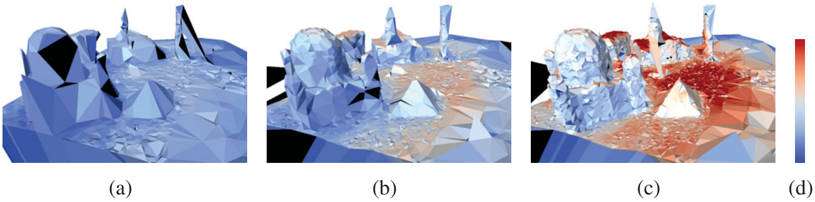


Figure 6: (a) - (c) The resulting mesh of the City-of-Sights after 10 (3204 points), 20 (8370 points) and 50 (19873 points) reconstructed images. (d) Colormap. Blue indicates a low number of cameras observing a triangle. Red indicates that a triangle is seen more than 30 times. Black triangles are captured less than 3 times. Best viewed in color.

7 Acknowledgement

This work has been supported by the Austrian Research Promotion Agency (FFG) FIT-IT project Construct (830035), Holistic (830044), and Pegasus (825841).

References

[1] A.Zisserman and R. Hartley. *Multi View Geometry*, pages 101–109. Cambridge University Press, 2000.

[2] J.-D. Boissonnat and M. Yvinec. *Algorithmic Geometry*. Cambridge University Press, Cambridge, U.K., 1998.

[3] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 23(11): 1222–1239, 2001.

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

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

[6] M. Golparvar Fard, F. Peña-Mora, and S. Savarese. Monitoring changes of 3D building elements from unordered photo collections. In *ICCV Workshops*, pages 249–256, 2011.

[7] G. Graber, T. Pock, and H. Bischof. Online 3D reconstruction using convex optimization. In *ICCV Workshops*, pages 708–711. IEEE, 2011.

[8] L. Gruber, S. Gauglitz, J. Ventura, S. Zollmann, M. Huber, M. Schlegel, G. Klinker, D. Schmalstieg, and T. Höllerer. The city of sights: Design, construction, and measurement of an augmented reality stage set. In *Proc. Nineth IEEE International Symposium on Mixed and Augmented Reality (ISMAR'10)*, pages 157–163, Seoul, Korea, Oct. 13-16 2010.

[9] H. Hirschmuller. Accurate and efficient stereo processing by semi-global matching and mutual information. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 807–814, 2005.

[10] C. Hoppe, A. Wendel, S. Zollmann, K. Pirker, A. Irschara, H. Bischof, and S. Kluckner. Photogrammetric camera network design for micro aerial vehicles. In *Proc. 17th Computer Vision Winter Workshop (CVWW)*, February 2012.

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

[12] A. Irschara, C. Zach, J. M. Frahm, and H. Bischof. From structure-from-motion point clouds to fast location recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[13] G. Klein and D.W. Murray. Parallel tracking and mapping for small AR workspaces. In *IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 225–234. IEEE, 2007.

[14] L. Kneip, D. Scaramuzza, and R. Siegwart. A novel parametrization of the perspective-three-point problem for a direct computation of absolute camera position and orientation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[15] P. Labatut, J.P. Pons, and R. Keriven. Efficient multi-view reconstruction of large-scale scenes using interest points, delaunay triangulation and graph cuts. In *International Conference on Computer Vision (ICCV)*, pages 1–8, 2007.

[16] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision (IJVC)*, 60(2):91–110, 2004.

[17] P. Mücke, R. Klowsky, and M. Goesele. Surface reconstruction from multi-resolution sample points. In *Vision, Modeling, and Visualization (VMV 2011)*, pages 105–112, 2011.

[18] A. R. Newcombe and A. J. Davison. Live dense reconstruction with a single moving camera. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1498–1505, 2010.

[19] D. Nistér. An efficient solution to the five-point relative pose problem. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 195–202, 2003.

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

[21] Q. Pan, G. Reitmayr, and T. Drummond. ProFORMA: Probabilistic Feature-based On-line Rapid Model Acquisition. In *Proc. 20th British Machine Vision Conference (BMVC)*, London, September 2009.

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

[23] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon. Bundle adjustment – A modern synthesis. In *Vision Algorithms: Theory and Practice*, pages 298–375. 2000.

[24] M. Trummer, C. Munkelt, and J. Denzler. Online next-best-view planning for accuracy optimization using an extended e-criterion. In *International Conference on Pattern Recognition (ICPR)*, pages 1642–1645, 2010.

[25] C. Wu. SiftGPU: A GPU implementation of scale invariant feature transform (SIFT). http://cs.unc.edu/˜ccwu/siftgpu, 2007.