REALISTIC 3D SCENE RECONSTRUCTION FROM UNCONSTRAINED AND UNCALIBRATED IMAGES TAKEN WITH A HANDHELD CAMERA

Minh Hoang Nguyen, Burkhard Wünsche, Patrice Delmas and Christof Lutteroth
Department of Computer Science, The University of Auckland, Auckland, New Zealand

Keywords: Structure-from-Motion, Image modelling, Fundamental matrix, RANSAC, SIFT, Image-based modelling, Surface reconstruction.

Abstract We address the problem of reconstructing 3D scenes from a set of unconstrained images. These image sequences can be acquired by a video camera or handheld digital camera without requiring calibration. Our approach does not require any a priori information about the cameras being used. The camera's motion and intrinsic parameters are all unknown. We use a novel combination of advanced computer vision algorithms for feature detection, feature matching, and projection matrix estimation in order to reconstruct a 3D point cloud representing the location of geometric features estimated from input images. In a second step a full 3D model is reconstructed using the projection matrix and a triangulation process. We demonstrate with data sets of different structures obtained under different weather conditions that our algorithm is stable and enables inexperienced users to easily create complex 3D content using a simple consumer level camera.

1 INTRODUCTION

The design of digital 3D scenes is an essential task for many applications in diverse fields such as architechture, engineering, education and arts. Traditional modelling systems such as Maya, 3D Max or Blender enable graphic designers to construct complicated 3D models via 3D meshes. However, the capability for inexperience users to create 3D models has not kept pace. Even for trained graphic designers with in-depth knowledge of computer graphics, constructing a 3D model using traditional modelling systems can still be a challenging task (Yang et al., 2010). Hence, there is a critical need for a better and more intuitive approach for reconstructing 3D scenes and models.

The past few years have seen significant progress toward this goal with the emergence of *structure from motion* (SFM) methods in the research community. There are two common approaches: laser scanners and image-based modelling approach. Laser scanners are very robust and highly accurate. However, they are very costly and have restrictions on the size and the surface properties of objects in the scene (Hu et al., 2008). In contrast, an image-based modelling approach reconstructs the geometry of a complex 3D scene from a sequence of images. The technique is usually less accurate, but offers a very intuitive and low-cost method for reconstructing 3D scenes and models.

We aim to create a low-cost system that allows users to obtain 3D reconstruction of a scene using an off-the-shelf handheld camera. The users accquire images by freely moving the camera around the scene. The system will then perform 3D reconstruction using the following steps:

1. Image Accquisition and Feature Extraction
2. Feature Matching
3. Fundamental Matrix and Projection Matrix Estimation
4. Bundle Adjustment and Refinement
5. Point Cloud Generation
6. Surface Reconstruction

The remainder of this paper is structured as follows. Section 2 disucsses relevant literature in the field. Section 3 presents our approach for reconstructing 3D scenes. Section 4 discusses our results. Section 5 concludes and summarises the paper and gives a brief outlook on directions for future research.

2 RELATED WORK

2.1 Image-based Modelling

Various image-based modelling techniques have

Nguyen M., Wünsche B., Delmas P. and Lutteroth C..
REALISTIC 3D SCENE RECONSTRUCTION FROM UNCONSTRAINED AND UNCALIBRATED IMAGES TAKEN WITH A HANDHELD CAMERA.
DOI: 10.5220/0003376900670075
In *Proceedings of the International Conference on Computer Graphics Theory and Applications* (GRAPP-2011), pages 67-75
ISBN: 978-989-8425-45-4

been explored in recent years. In this section, we discuss the most closely related work in image-based 3D reconstruction.

(Brown and Lowe, 2005) presented an image-based modelling system which aims to recover camera parameters, pose estimates and sparse 3D scene geometry from a sequence of images.

(Snavely *et al.*, 2006) presented the *Photo Tourism (Photosynth)* system which is based on the work of Brown and Lowe, with some significant modifications to improve scalability and robustness. (Schaffalitzky and Zisserman, 2002) proposed another related technique for calibrating unordered image sets, concentrating on efficiently matching points of interest between images. Although these approaches address the same SFM concepts as we do, their aim is not to reconstruct and visualise 3D scenes and models from images, but only to allow easy navigation between images in three dimension.

(Debevec *et al.*, 1996) introduced the Facade system for modelling and rendering simple architectural scenes by combining geometry-based and image-based techniques. The system requires only a few images and some known geometric parameters. It was used to reconstruct compelling fly-throughs of the Berkeley campus and it was employed for the MIT City Scanning Project, which captured thousands of calibrated images from an instrumented rig to compute a 3D model of the MIT campus. While the resulting 3D models are often impressive, the system requires input images taken from calibrated cameras.

(Hua *et al.,* 2007) tried to reconstruct a 3D surface model from a single uncalibrated image. The 3D information is acquired through geometric attributes such as coplanarity, orthogonality and parallelism. This method only needs one image, but this approach often poses severe restrictions on the image content.

(Criminisi *et al.*, 1999) proposed an approach that computes a 3D affine scene from a single perspective view of a scene. Information about geometry, such as the vanishing lines of reference planes, and vanishing points for directions not parallel to the plane, are determined. Without any prior knowledge of the intrinsic and extrinsic parameters of the cameras, the affine scene structure is estimated. This method requires only one image, but manual input is necessary.

2.2 Surface Reconstruction

Surface reconstruction from point clouds has been studied extensively in computer graphics in the past decade. A *Delaunay-based* algorithm proposed by (Cazals and Giesen, 2006) typically generates meshes which interpolate the input points. However, the resulting models often contain rough geometry when the input points are noisy. These methods often provide good results under prescribed sampling criteria (Amenta and Bern, 1998).

(Edelsbrunner *et al.*, 1994) presented the well-known *α-shape* approach. It performs a parameterised construction that associates a polyhedral shape with an unorganized set of points. A drawback of *α*-shapes is that it becomes difficult and sometimes impossible to choose *α* for non-uniform sampling so as to balance hole-filling against loss of detail (Amenta et al., 2001).

(Amenta *et al.*, 2001) proposed the *power crust* algorithm, which constructs a surface mesh by first approximating the medial axis transform (MAT) of the object. The surface mesh is then produced by using an inverse transform from the MAT.

Approximate surface reconstruction works mostly with implicit surface representations followed by iso-surfacing. (Hoppe *et al.*, 1992) presented a clean abstraction of the reconstruction problem. Their approach approximated the signed distance function induced by the surface F and constructed the output surface as a polygonal approximation of the zero-set of this function. Kazhdan *et al.* presented a method which is based on an implicit function framework. Their solution computes a 3D indicator function which is defined as 1 at point inside model and 0 as point outside model. The surface is then reconstructed by extracting an appropriate isosurface (Kazhdan et al., 2006).

3 METHODOLOGY

3.1 Feature Matching

The input for our reconstruction algorithm is a sequence of images of the same object taken from different views. The first step is to find feature points in each image. The accuracy of matched feature points affects the accuracy of the fundamental matrix and the computation of 3D points significantly. Many sophisticated algorithms have been proposed such as the Harris feature extractor (Derpanis. K, 2004) and the SUSAN feature extractor (Muyun et al., 2004). We use the SIFT (Scale Invariant Feature Transform) operator to detect, extract and describe local feature descriptors. Feature points extracted by SIFT are distinctive and invariant to different transformations, changes in illumination and have high information content (Hua et al., 2007) , (Brown et al., 2005).

The SIFT operator works by first locating potential keypoints of interest at maxima and minima of the result of the Difference of Gaussian (DoG) function in scale-space. The location and scale of each keypoint is then determined and keypoints are selected based on measures of stability. Unstable extremum points with low contrast and edge response features along an edge are discarded in order to accurately localise the keypoints. Each found keypoint is then assigned one or more orientations based on local image gradients. Finally, using local image gradients information, a descriptor is produced for each keypoint (Lowe et al., 1999).

Once features have been detected and extracted from all the images, they are matched. Since multiple images may view the same point in the world, each feature is matched to the nearest neighbours. During this process, image pairs whose number of corresponding features is below a certain threshold are removed. In our experiment, the threshold value of 20 seems to produce the best results.

The feature points matching between two images could be achieved by comparing each keypoint of the one image with keypoints of the other image. The Euclidean distance

$$D(A,B)=\|A-B\|=\sqrt{\sum^{\dim}\left(A_d-B_d\right)^2} \qquad (1)$$

is used to measure the similarity between two keypoints A and B. A small distance indicates that the two keypoints are close and thus of high similarity (Hu et al., 2008). However, a small Euclidean distance does not necessarily mean that the points represent the same feature. In order to accurately match a keypoint in the candidate image, we identify the closest and second closet keypoints in the reference image using a nearest neighbour search strategy. If the ratio of them is below a given threshold, the keypoint and the closest matched keypoint are accepted as correspondences, otherwise that match is rejected (Hu et al., 2008).

3.2 Image Matching

The next stage of our algorithm attempts to find all matching images. Matching images are those which contain a common subset of 3D points. From the feature matching stage, we have identified images with a large number of corresponding features. As each image could potentially match every other image, the problem may seem at first to be quadratic in the number of images. However, it has been shown by (Brown et al., 2005) that it is only neccessary to match each image to *k* neighbouring images in order to obtain a good solution for the image geometry. In our system, we use *k = 6*.

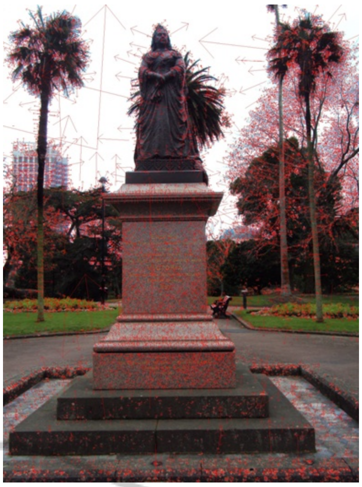

Figure 1: Feature Extraction - The red arrow symbol indicates the detected features. Detected features are displayed as vectors indicating scale, orientation and location.

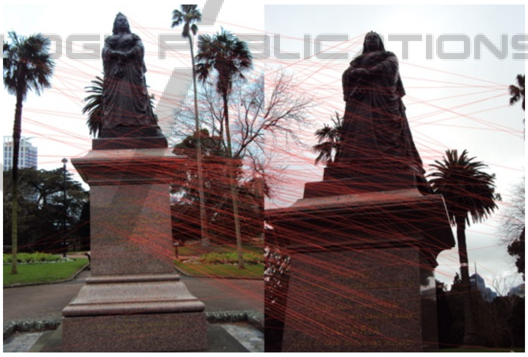

Figure 2: Matched Features.

3.3 Feature Space Outlier Rejection

We employ a feature space outlier rejection strategy that uses imformation from all of the images in the *n*-image matching context to remove incorrect matches. It has been shown that comparing the distance of a potential match to the distance of the best incorrect match is an effective strategy for outlier rejection (Brown et al., 2005).

The outlier rejection method works as follows: Assuming that there are *n* images which contain the same point in the world. Matches from these images are placed in an ordered list of nearest-neighbour matches. We assume that the first *n - 1* elements in the list are potentially correct, but the element *n* is incorrect. The distance of the n$^{\text{th}}$ element is denoted as outlier distance. We then verify the match by comparing the match distance of the potential correct match to the outlier distance. A match is only accepted if the match distance is less than 80% of the outlier distance, otherwise it is rejected. In general, the feature space outlier rejection test is very effective and reliable. For instance, a substantial number of the false matches (up to 80%) can be simply eliminated for a loss of less than 10% of correct matches. This allows for a significant reduction in the number of RANSAC iterations required in subsequent steps (Brown et al., 2005).

3.4 Fundamental Matrix Estimation

At this stage, we have a set of putative matching image pairs, each of which shares a set of individual correspondences. Since our matching procedure is only based on the similarity of keypoints, it inevitably produces mismatches. Many of matches will therefore be spurious. Fortunately, it is possible to use a geometric consistency test to eliminate many of these spurious matches. The epipolar geometry of a given image pair can be expressed using the fundamental matrix *F*.

For each remaining pair of matching images, we use their corresponding features to estimate the fundamental matrix. This geometric relationship of a given image pair can be expressed as

$$u^T F v = 0 \tag{2}$$

for any pair of matching features $u \leftrightarrow v$ in the two images. The coefficients of the equation (2) can be written in terms of the known coordinates *u* and *v*.

$$x' x f_{11} + x' y f_{12} + x' f_{13} + y' x f_{21} + y' y f_{22} + y' f_{23} + x f_{31} + y f_{32} + f_{33} = 0$$

$$\Leftrightarrow (x' x, x' y, x', y' x, y' y, y', x, y, 1) f = 0$$

$$u = (x', y', 1) \text{ and } v = (x, y, 1)$$

where

$$f = [f_{11}, f_{12}, f_{13}, f_{21}, f_{22}, f_{23}, f_{31}, f_{32}, f_{33}]$$

From a set of *n* correspondent points, we can obtain a set of linear equations of the form

$$Af = \begin{bmatrix} x_1' x_1 & x_1' y_1 & x_1' & y_1' x_1 & y_1' y_1 & y_1' & x_1 & y_1 & 1 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n' x_n & x_n' y_n & x_n' & y_n' x_n & y_n' y_n & y_n' & x_n & y_n & 1 \end{bmatrix} f = 0$$

Thus a unique solution of *F* (up to scale) can be determined if we are given 8 correspondences (Hartley et al., 2003). Usually considerable more than 8 correspondences are used because of inaccuracies in the feature estimates. The resulting overdetermined system can be solved resulting in a solution optimal in a least squares sense, which is then used to compute the fundamental matrix.

Many solutions have been proposed to estimate the fundamental matrix. In our system, we use *RANSAC* (Hartley et al., 2003) to robustly estimate *F*. Inside each iteration of RANSAC, the 8-point algorithm, followed by non-linear estimation step, is used to compute a fundamental matrix (Hartley et al., 2003). The computed epipolar geometry is then used to refine the matching process.

3.5 Bundle Adjustment

Next, given a set of geometrically consistent matches between images, we need to compute a 3D camera pose and scene geometry. This step is critical for the accuracy of the reconstruction, as concentration of pairwise homographies would accumulate errors and disregard constrains between images. The recovered geometry parameters should be consistent. That is, the reprojection error, which is defined by the distance between the projections of each keypoint and its observations, is minimised (Brown et al., 2005).

This error minimization problem can be solved using *Bundle Adjustment*. Bundle Adjustment is a well-known method of refining a visual reconstruction to produce joinly optimal 3D structure and viewing parameter estimates. It attemps to minimise the reprojection error between observed and predicted image points, which is expressed as the sum of squares of a number of non-linear real-valued functions (Brown et al., 2005).

$$e = \sqrt{\frac{1}{NM} \sum_{i=1}^{N} \sum_{j=1}^{M} \left[(\tilde{x}_{ij} - x_{ij})^2 + (\tilde{y}_{ij} - y_{ij})^2 \right]} \tag{3}$$

where $p = (x_{ij}, y_{ij})$ denotes the coordinate of an image point, and $\tilde{p} = (\tilde{x}_{ij}, \tilde{y}_{ij})$ denotes the observed image point.

The minimization can be formulated as a non-linear least squares problem and solved with algorithms such as Levenberg-Marquardt (LM). Such algorithms are particularly prone to bad local minima, so it is important to provide a good initial estimate of the parameters (Snavely et al., 2006).

The bundle adjustment algorithm starts by selecting an initial image pair, which has a large number of matches and a large baseline. This is to ensure that the location of the 3D observed point is well-conditioned. The bundle adjustment algorithm will then estimate geometry parameters for the given pair. Subsequent images are added to the bundle adjuster one at a time, with the best matching (maximum number of matched) image being added at each step. Each image is initialised with the same rotation and focal length as the image to which it best matches. This has proved to work very well even though images have different rotation and scale (Snavely et al., 2006), (Brown et al., 2005).

Figure 3 shows the original model of the Daliborka tower and its generated point clouds.

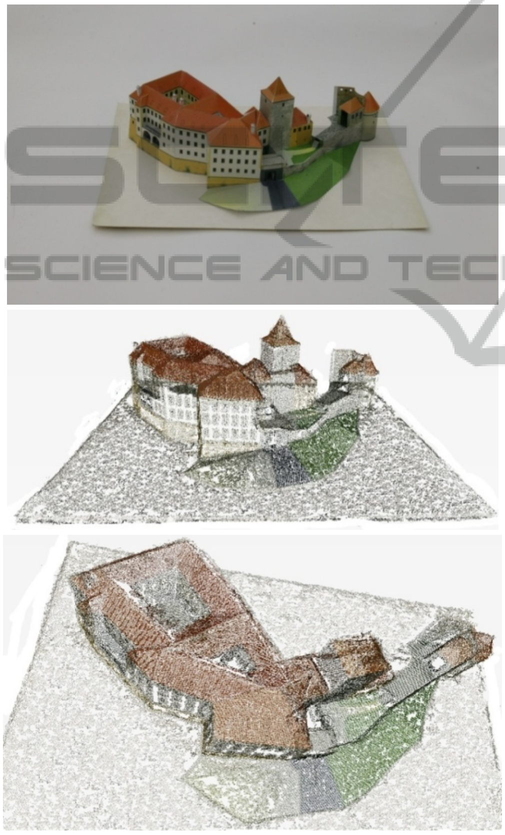

Figure 3: Model of the Daliborka tower (3D Reconstruction Dataset. Centre for Machine Perception) and its generated point clouds.

3.6 Surface Reconstruction

The final step is to reconstruct surfaces from the obtained point clouds. Our objective is to find a piecewise linear surface that closely approximates the underlying 3D models from which the point clouds was sampled (Kazhdan et al., 2006). Many sophisticated surface reconstructions have been proposed and extensively studied. In our system, we employ the *Power Crust* algorithm (Amenta et al., 2001) for remeshing the surfaces.

The Power Crust algorithm reconstructs surfaces by first attempting to approximate the medial axis transform of the object. The surface representation of the point clouds is then produced by the inverse transform. The algorithm is composed of 4 simple steps: 1) A 3D Voronoi diagram is computed from the sample points. 2) For each point *s*, select the furthest vertex v_1 of its Voronoi cell, and the furthest vertex v_2 such that the angle $v_1 s v_2$ is greater than 90 degree. 3) Compute the Voronoi diagram of the sample point and the Voronoi vertices selected from the second stage. 4) Create Delaunay triangulation from the Voronoi diagram in the previous stage. An example of the resulting 3D model is illustrated in figure 4. The complete algorithm is summarised in figure 5.

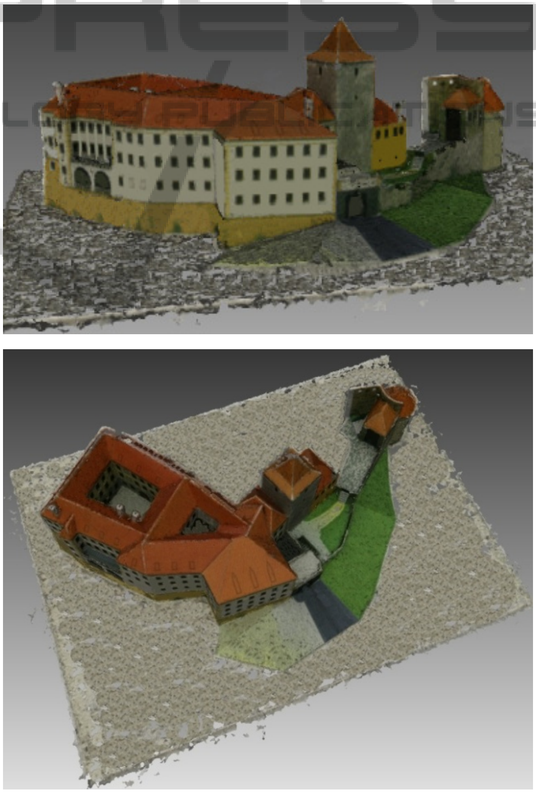

Figure 4: The reconstruction of the model of the Daliborka tower in Figure 3.

```
Algorithm for 3D Object Reconstruction
Input: n unordered and unconstrained images
    1. Extract features from all input images using SIFT operator
    2. Find t nearest neightbors for each feature
    3. For each image:
        a. Select k candidate matching images (those which have highest number of features matched to this image)
        b. Find geometrically consistent feature matches using RANSAC to solve for fundamental matrix between pairs of images.
    4. Compute 3D camera pose and scene geometry using Bundle Adjustment.
    5. Reconstruct surface for the obtained point clouds.
    6. (Future work) Apply hole-filling alogorithm for the resulting model.
Output: 3D model of the object
```

Figure 5: Algorithm for 3D Object Reconstruction.

4 RESULTS

We have tested our system with a number of different datasets, both indoor and outdoor scenes. In all our test cases, the system produces good results for rough, non-uniform and full-of-feature datasets. Datasets with smooth and uniform surfaces often result in inadequate number of 3D points generated, since the feature detector (SIFT) has trouble detecting and extracting features from these images. The size of our test datasets varies from as few as 6 images to hundreds of images, which are all taken with a simple handheld camera.

Dataset 1

The first data set consists of 37 images taken from arbitrary view directions on ground level using a normal consumer-level SONY DSC-W180 camera. The reconstructed 3D model has 19568 faces and is of good quality. The original object can be easily identified. Some holes exist near concave regions and near sharp corners. This is caused by large variations in the point cloud density, which the surface reconstruction algorithm was unable to deal with.

Dataset 2

The second data set comprises 55 images taken at ground level from two sides of the Saint Benedict Church in Auckland, New Zealand. The other two sides were not accessible. The images were taken with the same camera as in the previous case and under slightly rainy conditions. The reconstruction results are satisfactory. The resulting model which is composed of 37854 faces has a high resemblance with the original object and even the inaccessible sides look plausible. A few details, such as some windows, are missing causing holes in the model.

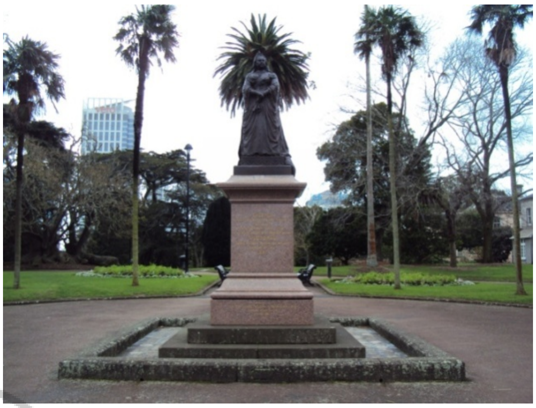

Figure 6.1: The statue of Queen Victoria, Mt Albert Park, Auckland - Original view.

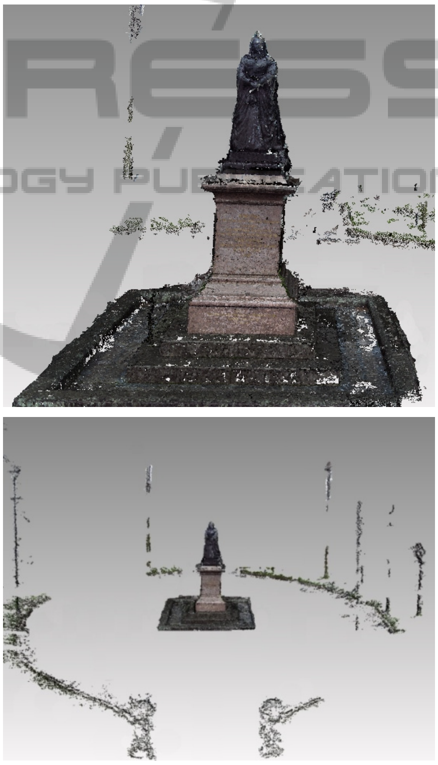

Figure 6.2: Two views of the reconstructed model of the statue of Queen Victoria. Number of images: 37 (2592x1944). Running time: approximately 4 hours.

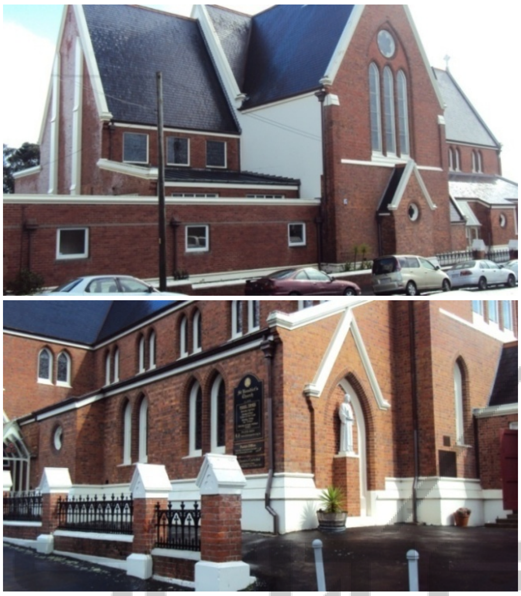

Figure 6.3a: Saint Benedict Church, Auckland.

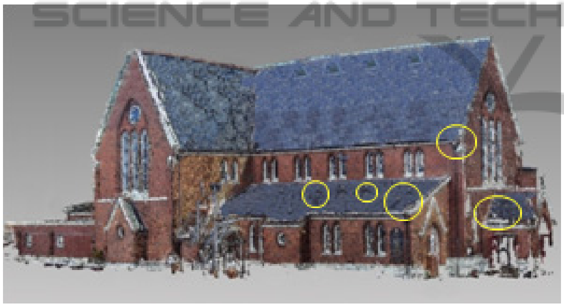

Figure 6.3b: Reconstructed model of Saint Benedict Church. The yellow circle indicates a reconstructed region which was invisible in all input images. Number of images: 55 (3648x2056). Running time: approximately 6h40 hours.

Dataset 3

The third data sets consisted of 63 images of Saint George church. All images were taken from ground level. Since the roof of that building is quite flat, this resulted in missing information about the roof structure and the reconstructed model contains large gaps in that area. We intend to overcome this type of problems with a sketch-based interface, which allows the users to add missing geometric details. The model contains of 28846 faces.

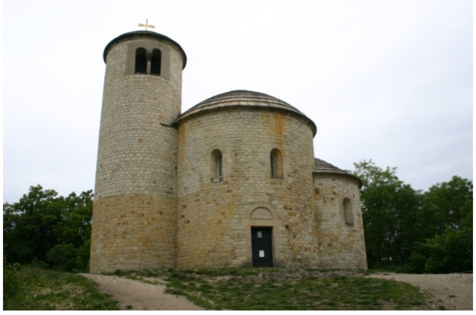

Figure 6.4a: Saint George (3D Reconstruction Dataset. Centre for Machine Perception). Input images: 63 (2048x3072).

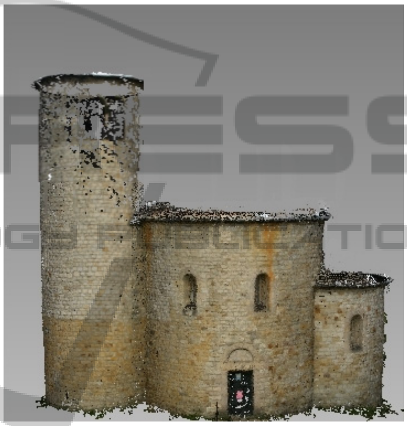

Figure 6.4b: Reconstructed model of Saint George Church. Number of images: 63 (2048x3072). Running time: approximately 9 hours.

Dataset 4

The fourth data set comprises 65 images taken from many different views of the model of the Daliborka tower shown in figure 3. The reconstruction result is of very good quality and the final model has a high resemblance with the original object. Small details such as windows are also properly reconstructed. The improved reconstruction is probably due to less geometric features in the original model and a more even illumination compared to outdoor scenes. The resulting model is composed of 29768 polygons. The computation time of this data set is over 9 hours. Figure 7 summarizes the computation time and parameters of the input data sets and resulting 3D models for the presented examples. It can be seen that the computation is quite slow, however, since it can be performed in an offline process, this is acceptable for our purpose.

Dataset	Statue of Queen Victoria	Saint Benedict Church	Saint George Church	Daliborka Tower
Number of Images	37	55	63	65
Image Resolution	2592x1944	3648x2056	2048x3072	4064x2704
Computation time in hour	4.1	6.4	9.0	> 9 .0
Generated Polygon	19568	37854	28846	29768

Figure 7: Comparison of the running time for reconstructing 3D models from different input data sets (photos). All examples were executed on a machine with an Intel Quad-Core i7 and 6GB RAM.

5 CONCLUSIONS AND FUTURE WORK

In this paper, we have discussed a novel approach for reconstructing realistic 3D models from a sequence of unconstrained and uncalibrated images. Geometry parameters such as cameras' pose are estimated automatically using a bundle adjustment method. 3D point clouds are then obtained by triangulation using the estimated projection matrix. We reconstruct surfaces for the point clouds to recover the original model. In contrast to previous approaches, we acquired the input images in just a few minutes with a simple hand-held consumer level camera. Our results demonstrate that our algorithm enables inexperienced users to easily create complex 3D content using a simple consumer level camera. This significantly simplifies the content creation process when constructing virtual environments. Problems, such as holes, still exist with the resulting model. This is caused by large variation in the point cloud's density. Another disadvantage is that the computation is quite expensive (the system takes over 4 hours to process 37 images, and about 9 hours for 63 images on a Intel Quad Core i7 with 6GB RAM), but this is only an issue in applications where the user needs the content immediately. A common problem with this application is that not all views of a model are obtainable. Especially the roof is often not fully or not at all visible. Similarly in some cases the backside of a building or object might not be accessible. We propose to use sketch input and symmetry information to "complete" models in such circumstances. Additional future work will concentrate on improved hole filling algorithms and on speeding up the algorithm by using an GPU implementation.

REFERENCES

Yang. R and Wünsche. B. (2010). Life-Sketch - A Framework for Sketch-Based Modelling and Animation of 3D Objects. AUIC '10 *Proceedings of the Eleventh Australasian Conference on User Interface*, Volume 106, pp. 1-9.

Hu. S, Qiao. J, Zhang. A and Huang. Q. (2008). 3D Reconstruction from Image sequence taken with a handheld camera. *International Society for Photogrammetry and Remote Sensing*, Vol 37, pp. 1-4.

Zhang. J, Boutin. M, Aliaga. D. G. Robust Bundle Adjustment for structure from motion. *Image Processing, 2006 IEEE International Conference*, pp. 2185-2188.

Snavely. N, Seitz. M. S, Szeliski. R. (2006). Photo tourism: Exploring photo collections in 3D. *ACM Transactions on Graphics - SIGGRAPH Proceedings*, 25(3), 2006, pp. 835-846.

Debevec. P, Taylor. C and Malik. J. Modeling and Rendering Architecture From Photographs: A Hybrid Geometry and Image-Based Approach. *In SIGGRAPH*, 1996, pp. 11-20.

Criminisi. A, Reid. I and Zisserman. A. Single View Metrology. *International Journal of Computer Vision (2000)*, pp. 123-148 .

Hua. S and Liu.T. Realistic 3D Reconstruction from Two Uncalibrated Views. *International Journal of Computer Science and Network Security, Volume 7 No.6 (2007)*, pp. 178-183.

Amenta. N, Choi. S, Kolluri. R. K. The Power Crust. *International Journal* of Computational Geometry: Theory and Applications (2000), Volume 19, pp. 127-153.

Kazhdan. M, Bolitho. M, Hoppe. H. Poisson Surface Reconstruction. *ACM International Conference Proceeding Series*; Vol. 256 (2006), pp. 185-192.

Triggs. B, McLauchlan. P. F, Hartley. R. I, Fitzgibbon. A. W. Bundle Adjustment - A Modern Synthesis. ICCV '99: *Proceedings of the International Workshop on Vision Algorithms. Springer-Verlag*. pp. 298–372.

Brown. M, Lowe. D. G. Unsupervised 3D Object Recognition and Reconstruction in Unordered Datasets. 3-D *Proceedings of the Fifth International Conference on Digital Imaging and Modelling*, 2005. pp. 110-119.

Kolmogorov, V and Zabih. R. Multi-camera Scene Reconstruction via Graph Cuts. In European Conference on Computer Vision (ECCV), May 2002. pp. 82-96.

Derpanis. G. K. The Harris Corner Detector. 2004. URL: http://www.cse.yorku.ca/~kosta/CompVis_ Notes/harris_detector.pdf

Muyun. W and Mingyi. H. Image Feature Detection and Matching Based on SUSAN Method. First

International Conference on Innovative Computing, Information and Control - Volume I (ICICIC'06), 2006. pp. 322-325.

Lowe, D. G. Object recognition from local scale-invariant features. *Proceedings of the International Conference on Computer Vision*. 2. pp. 1150–1157. 1999.

Choudhury. R. Recognizing pictures at an exhibition using SIFT. Biomedical Infomatics Department, Stanford University, USA. EE 368 Project Report, 2007. URL: http://www.stanford.edu/class/ee368/Project_07/reports/ee368group11.pdf

Brown. M, Szeliski. R and Winder. S. Multi-Image Matching using Multi-Scale Oriented Patches. *In Proceedings of the Interational Conference on Computer Vision and Pattern Recognition*, San Diego, June 2005. Vol 1, pp. 510-517.

Hartley. R and Zissweman. A. MultipleView Geometry in computer vision. Cambridge Press. 2003.

Kazhdan. M, M. Bolitho, H. Hoppe. Poisson Surface reconstruction. Symposium on Geometry Processing 2006, pp. 61-70.

Hoppe. H, DeRose. T, Duchamp, T. McDonald, J. Stuetzle. W. Surface reconstruction from unorganized points. *ACM SIGGRAPH 1992 Conference Proceedings*, pp. 71-78.

Hoppe. H, DeRose. T, Duchamp, T. McDonald, J. Stuetzle. W.. Mesh optimization. *ACM SIGGRAPH 1993 Conference Proceedings*, pp. 19-26.

Edelsbrunner. H and Mucke. E. P. Three-dimensional alpha shapes. ACM Trans. Graphics 13 (1994), pp. 43-72.

Edelsbrunner. H. Surface reconstruction by wrapping finite point sets in space. Discrete and Computational Geometry. The Goodman-Pollack Festschrift, ed. B. Aronov, S. Basu, J. Pach and M. Sharir, Springer-Verlag, 2003, pp. 379-404.

Cheng. H.L, Dey. T.K, Edelsbrunner. H and Sullivan. J. Dynamic skin triangulation. Discrete Comput. Geom. 25 (2001), pp. 525-568.