

Modelling of 3D Objects Using Unconstrained and Uncalibrated Images Taken with a Handheld Camera

Minh Hoang Nguyen¹, Burkhard Wünsche¹, Patrice Delmas², and Christof Lutteroth¹

¹ University of Auckland, Dept. of Computer Science, Graphics Group, Private Bag 92019, Auckland, New Zealand

² University of Auckland, Dept. of Computer Science, Intelligent Vision Group, Private Bag 92019, Auckland, New Zealand

hngu039@aucklanduni.ac.nz, burkhard@cs.auckland.ac.nz,
lutteroth@cs.auckland.ac.nz, p.delmas@auckland.ac.nz

Abstract. 3D models are an essential part of computer graphics applications such as games, movie special effects, urban and landscape design, architecture, virtual heritage, visual impact studies, and virtual environments such as *Second Life*. We have developed a novel technique which allows the construction of 3D models using image sequences acquired by a handheld low-cost digital camera. In contrast to alternative technologies, such as laser scanners, structured lighting, and sets of calibrated cameras, our approach can be used by everyone having access to a consumer-level camera. The user only has to create a set of images from different view directions, input them into our algorithm, and a 3D model is returned. 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 tested our algorithm using a variety of data sets of objects of different scales acquired under different weather and lighting conditions. The results show 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 architecture, 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 inexperienced 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 [25]. Hence, there is a critical need for a better and more intuitive approach for reconstructing 3D scenes and models.

The past decade has seen significant progress toward this goal. There are three common approaches: laser scanners, structured lighting, 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). Structured lighting is cheaper and works for a wide variety of materials, but is less accurate and restricts the size and distance of imaged objects.

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. Users acquire images by freely moving the camera around the scene. The system will then perform 3D reconstruction using the following steps:

1. Image Acquisition 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 discusses 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 section 6 gives a brief outlook on directions for future research.

2 Related Work

2.1 Image-Based Modelling

The reconstruction of 3D models from 2D images of objects has long been a major research field in Computer Vision. If the object geometry is relatively smooth and the material homogeneous, then surface geometry can be estimated from illumination gradients. This class of algorithm is called “shape from shading” [27]. However, the technique has too many constraints with respect to the object geometry and the environment in order to be useful in general applications.

A different approach is to only consider the silhouette of an object from different view points. This “shape from silhouette” class of algorithms is very fast and stable with regard to object colour, texture and material, but is very limited in the object geometries it can handle [20, 11].

A more stable and general approach with respect to object geometry is “structured lighting”, which projects a stable pattern of light onto a surface and reconstructs geometry from its deformation [7]. When combined with a camera both object geometry and texture can be recovered. This approach is used by the popular *Kinect* controller [28]. However, the device is not as portable as a camera, and the type of objects which can be reconstructed is constrained, since structured lighting only works within a limit range of distances and object sizes.

More general shape reconstruction algorithms can be developed by analysing features in camera images and video data. This class of techniques has usually less limitations, since cameras are designed to work for objects of vastly different scale, within a large range of distances, and for both indoor and outdoor environments. Furthermore, the technique can be used by a wide number of users, since cameras are readily available, very portable, and exist already in a large number of consumer devices such as mobile phones.

Brown and Lowe [4] 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. [24] 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 [22, 23] 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. [8] 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.

The work by Martinec and Pajdla [19] is closest to our research in the sense that the authors use arbitrary unconstrained input images. The solution is optimised to deal with cases where information is sparse, e.g., where some features occur on only two images. The given reconstruction results, especially for buildings, seem to be inferior to our method, but a comparison is difficult without using the same input data sets.

Hua et al. [16] 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. [6] 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

Image-based reconstruction methods match features in images and estimate their 3D position. The reconstruction of surfaces from such “point clouds” has been studied extensively in computer graphics in the past decade. A Delaunay-based algorithm pro-

posed by Cazals and Giesen [5] 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 [1].

Edelsbrunner et al. presented the well-known α -shape approach [10]. 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 [2].

Amenta et al. proposed the power crust algorithm [2], 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.

Approximating surface reconstruction methods often use an implicit surface representations followed by iso-surfacing. Hoppe et al. [14] 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 for points inside the model and as 0 for points outside the model. The surface is then reconstructed by extracting an appropriate isosurface [17].

3 Design

The main steps of our solution are summarised in figure 1. Each step is explained in more detail in the subsequent subsections.

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 [12, 9] and the SUSAN feature extractor [21]. We use the SIFT (Scale Invariant Feature Transform) operator to detect, extract and describe local feature descriptors [18]. Feature points extracted by SIFT are distinctive and invariant to different transformations, changes in illumination and have high information content [16, 4].

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 gradient information, a descriptor is produced for each keypoint [18]. Figure 2 shows an example.

Algorithm for 3D Object Reconstruction**Input:** n unordered and unconstrained images

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

Output: 3D model of the object

Fig. 1. Overview of our algorithm for reconstructing 3D models from a set of unconstrained and uncalibrated images.

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 matching of features between two images can be achieved by comparing the spatial relationship between keypoints. The Euclidean distance

$$D(A, B) = \|A - B\|_2 = \sqrt{\sum_{i=1}^{dim} (A_i - B_i)^2}$$

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 [15]. 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 closest 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 [15].

An example demonstrating detected matching features is given in figure 3. Note that for some regions neighbouring features are matched to completely different features in the second input image, whereas for other regions matched features have a consistent spatial relationship.



Fig. 2. Extracted features displayed with red arrows indicating scale, orientation and location.

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 [4] that it is only necessary to match each image to k neighbouring images in order to obtain a good solution for the image geometry. In our system, we use $k = 6$.

3.3 Feature Space Outlier Rejection

We employ a feature space outlier rejection strategy that uses information 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 [4].



Fig. 3. Matching features between two images indicated by red lines. Note that in the initial matching step the majority of matches are spurious and often refer to completely different objects, such as the top of the statue and the tip of a streetlight in the above pictures.

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^{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 [4].

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 spurious matches. We eliminate many of these spurious matches by using a geometric consistency test, which is based on the epipolar geometry of a given image pair expressed using the fundamental matrix \mathbf{F} .

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

pressed as

$$\mathbf{u}^T \mathbf{F} \mathbf{v} = 0 \quad (1)$$

for any pair of matching features $\mathbf{u} \leftrightarrow \mathbf{v}$ in the two images. The coefficients of the equation 1 can be written in terms of the known coordinates $\mathbf{u} = (x', v', 1)$ and $\mathbf{v} = (x, y, 1)$:

$$\begin{aligned} & x'x f_{11} + x'y f_{12} + x' f_{13} + y'x f_{21} + y'y f_{22} + y' f_{23} + x f_{31} + y f_{32} + f_{33} = 0 \\ \Leftrightarrow & (x'x, x'y, x', y'x, y'y, y', x, y, 1) \mathbf{f} = 0 \end{aligned}$$

where

$$\mathbf{f} = (f_{11}, f_{12}, f_{13}, f_{21}, f_{22}, f_{23}, f_{31}, f_{32}, f_{33})^T$$

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

$$\mathbf{A} \mathbf{f} = \begin{pmatrix} x'_1 x_1 & x'_1 y_1 & x'_1 & y'_1 x_1 & y'_1 y_1 & y'_1 & x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x'_n x_n & x'_n y_n & x'_n & y'_n x_n & y'_n y_n & y'_n & x_n & y_n & 1 \end{pmatrix} \mathbf{f} = 0$$

Thus a unique solution of \mathbf{F} (up to scale) can be determined if we are given 8 correspondences [13]. 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 [13] to robustly estimate \mathbf{F} . Inside each iteration of RANSAC, the 8-point algorithm, followed by anon-linear estimation step, is used to compute a fundamental matrix [13]. 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 [4].

This error minimization problem can be solved using *Bundle Adjustment*, which is a well-known method of refining a visual reconstruction to produce jointly optimal 3D structure and viewing parameter estimates. It attempts to minimise the reprojection error e between observed image points $\tilde{\mathbf{p}} = (\tilde{x}_{ij}, \tilde{y}_{ij})$ and predicted image points $\mathbf{p} = (x_{ij}, y_{ij})$. This error is expressed [4]:

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

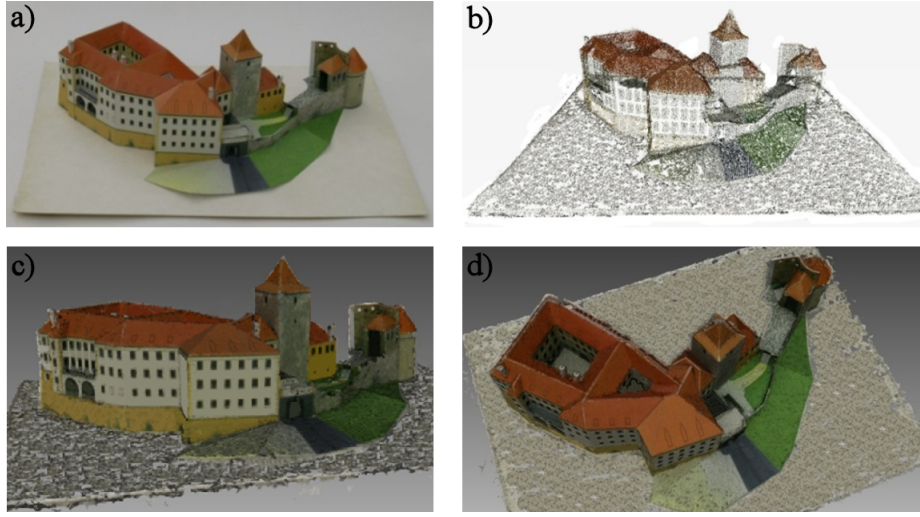


Fig. 4. An example of the surface reconstruction process: (a) The original object, a paper model of the Daliborka tower at Prague castle [3]; (b) The point cloud resulting from matching features in images and estimating their 3D position using bundle adjustment; and (c,d) the 3D model obtained by reconstructing the surface represented by the point cloud.

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 [24].

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 image (maximum number of matching features) 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 [26, 24, 4]. Figure 4 (a) shows one image of a set of images of a paper model of the Daliborka tower at Prague castle [3]. Part (b) of the figure shows the point cloud resulting from matching features in images and estimating their 3D position using bundle adjustment.

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 [17]. Many sophisticated surface reconstructions have been proposed and extensively studied. In our system, we employ the Power Crust algorithm [2] for remeshing the surface.



Fig. 5. An image of the statue of Queen Victoria in Mt Albert Park, Auckland (left) and the 3D model reconstructed from 37 images of the statue with resolution 2592x1944 (right). The computation time was about 4 hours.

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 \mathbf{s} , select the furthest vertex \mathbf{v}_1 of its Voronoi cell, and the furthest vertex \mathbf{v}_2 such that the angle $\mathbf{v}_1\mathbf{s}\mathbf{v}_2$ 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 a Delaunay triangulation from the Voronoi diagram in the previous stage.

Figure 4 (c) and (d) show the result of applying the Power Crust algorithm to the point cloud in part (b) of the figure.

4 Results

We have tested our system with a number of different datasets, including indoor and outdoor scenes and objects of different scale. In all of our test cases, the system produces good results for rough, non-uniform and feature-rich 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. Note that instead of many individual images the user can also create a short video sequence.

4.1 Dataset 1 - Statue of Queen Victoria

The first test data set consists of 37 images of the statue of Queen Victoria in Mt Albert Park, Auckland. The images were taken from arbitrary view directions on ground



Fig. 6. Two images of the Saint Benedict Church in Auckland, New Zealand (top) and the 3D model reconstructed from 55 images of the church with resolution 2592x1944 (bottom). The computation time was about 6 1/2 hours.

level using a normal consumer-level SONY DSC-W180 camera. The reconstructed 3D model, shown in figure 5 on the right, has 19568 faces and is of moderate 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.

4.2 Dataset 2 - Saint Benedict Church

The second test data set comprises 55 images of Saint Benedict Church in Auckland, New Zealand. The photos were taken at ground level from only two sides of the church. The other two sides were not accessible. The images were taken with the same camera as in the previous example and under slightly rainy conditions. The reconstruction results are satisfactory as illustrated in the image on the bottom of figure 6. The resulting 3D model 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 inaccurately reconstructed windows, are causing holes in the model.

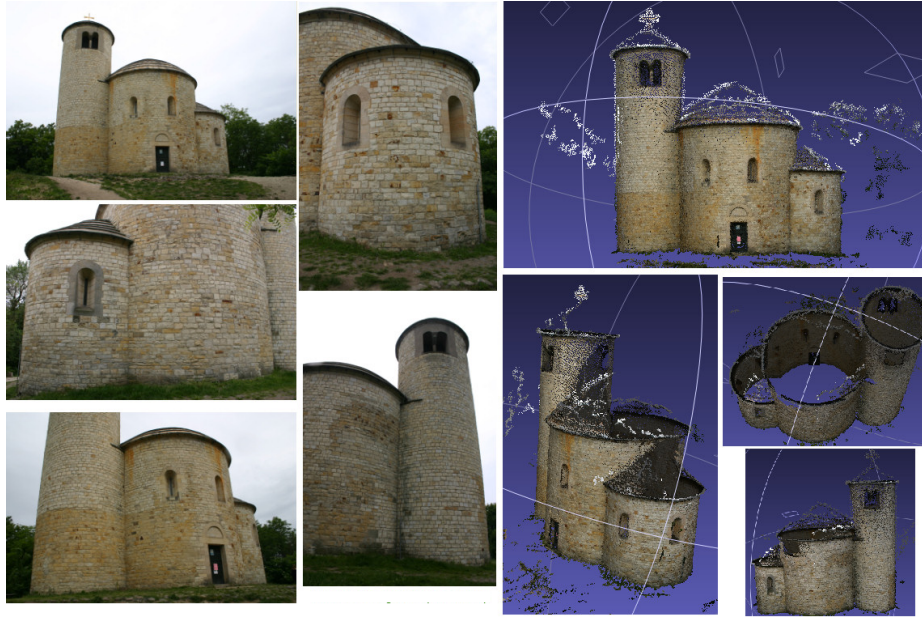


Fig. 7. Image of the Saint George Church (left) and the 3D model reconstructed from 63 images of the church with resolution 2048x3072 (right). The computation time was about 9 hours.

4.3 Dataset 3 - Saint George Church

The third test data set consisted of 63 images of Saint George church. All images were taken from ground level. Since the roof of that building is quite flat, few images show the roof area. Consequently a large part of the area is not covered by any images, and for the remaining parts of the roof feature matching becomes very difficult. Figure 7 illustrates that this results in large gaps in the roof region. In future work we intend to overcome this type of problems with a sketch-based interface, which allows the users to add missing geometric details. The reconstructed model contains 28846 faces.

4.4 Dataset 4 - Paper model of the Daliborka Tower at Prague castle

The fourth test data set comprises 65 images taken from many different views of a paper model of the Daliborka tower at Prague castle [3]. Figure 4 shows that the reconstruction 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 due to more pronounced and less complex features, a smaller number of features, 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.



Fig. 8. Three images of a small bunny model (left) and the resulting reconstruction (bottom) using 46 input images with a resolution of 2592 x 1944 pixels.

4.5 Dataset 5 - Small Bunny Model

The last test data set consists of 46 images of a small bunny model with a resolution of 2592 x 1944 pixels. The images were taken with a consumer-level SONY DSC-W180 camera, and covered all view directions. The original model has a very bumpy surface, which is extremely difficult to reconstruct. Figure 8 shows that the reconstruction is of good quality and has a high resemblance to the original object. One visible artifact is the difference in texture compared to the original model. This is due to the large brightness differences between pixels representing the same surface point in different images.

4.6 Comparison

Table 1 shows a comparison of the results of our algorithm for different input data sets. All computations were performed on an Intel Quad Core i7 with 6GB RAM. The computation time for the presented data sets varies from 4 hours to more than 9 hours. It is largely dependent on the number and size of images, but also the complexity and material properties of the objects in the scene, which influence the number of features detected in each image. Reconstruction quality depends on the ability of the algorithm to correctly match corresponding features in different images and an adequate coverage of all areas of an object in the input data set. A common practical limitation is the top of large objects such as buildings.

Data set	Statue of Queen Victoria	Saint Benedict Church	Saint George Church	Daliborka Tower	Bunny Figure
Number of Images	37	55	63	65	64
Image Resolution	2592×1944	3648×2056	2048×3072	4064×2704	2592×1944
Computation time	4.1 hours	6.4 hours	9.0 hours	>9.0 hours	4 hours
Number of polygons	19568	37854	28846	29768	33858
Artifacts	Lack of surface details, holes	Roof texture not smooth, Small details missing	Roof missing, holes	Ground plane poorly represented	Surface texture inaccurate

Table 1. Comparison of reconstruction results for different input data sets.

5 Conclusion

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. Advantages of our approach are scalability (permitted size and distance of objects), high stability with regard to different environment conditions, and ease-of-use and affordability (a consumer level camera is sufficient).

Problems, such as holes, still exist with the resulting 3D models. This is caused by large variation in the point cloud’s density. The algorithm also has problems for large areas without features (e.g. flat uni-coloured surfaces) and regions with rough surface geometry. 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.

6 Future Work

We propose to use sketch input and symmetry information to “complete” regions not properly reconstructed in the 3D model due to lack of coverage on the input images. Additional future work will concentrate on improved hole filling algorithms and on speeding up the algorithm by using an GPU implementation.

References

1. AMENTA, N., AND BERN, M. Surface reconstruction by voronoi filtering. In *Proceedings of the fourteenth annual symposium on Computational geometry* (New York, NY, USA, 1998), SCG '98, ACM, pp. 39–48.
2. AMENTA, N., CHOI, S., AND KOLLURI, R. K. The power crust. In *Proceedings of the sixth ACM symposium on Solid modeling and applications* (New York, NY, USA, 2001), SMA '01, ACM, pp. 249–266.
3. BETEXA ZS S.R.O. The prague castle, paper model of the biggest castle complex in the world, scale 1:450, 2006.
4. BROWN, M., AND LOWE, D. G. Unsupervised 3d object recognition and reconstruction in unordered datasets. In *Proceedings of the Fifth International Conference on 3-D Digital Imaging and Modeling* (Washington, DC, USA, 2005), IEEE Computer Society, pp. 56–63.
5. CAZALS, F., AND GIESEN, J. Delaunay triangulation based surface reconstruction: Ideas and algorithms. In *Effective Computational Geometry for Curves and Surfaces* (2006), Springer, pp. 231–273.
6. CRIMINISI, A., REID, I., AND ZISSERMAN, A. Single view metrology. *International Journal of Computer Vision* 40 (November 2000), 123–148.
7. DAVID FOI, T. S., AND VOISIN, Y. A comparative survey on invisible structured light. In *Proceedings of SPIE, volume 5303* (2004), pp. 90–98.
8. DEBEVEC, P. E., TAYLOR, C. J., AND MALIK, J. Modeling and rendering architecture from photographs: a hybrid geometry- and image-based approach. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques* (New York, NY, USA, 1996), SIGGRAPH '96, ACM, pp. 11–20.
9. DERPANIS, K. G. The harris corner detector, 2004. http://www.cse.yorku.ca/~kosta/CompVis_Notes/harris_detector.pdf.
10. EDELSBRUNNER, H., AND MÜCKE, E. P. Three-dimensional alpha shapes. *ACM Trans. Graphics* 13, 1 (Jan. 1994), 43–72.
11. FRANCO, J.-S., LAPIERRE, M., AND BOYER, E. Visual shapes of silhouette sets. *3D Data Processing Visualization and Transmission, International Symposium on 0* (2006), 397–404.
12. HARRIS, C., AND STEPHENS, M. A combined corner and edge detector. In *Proceedings of the 4th Alvey Vision Conference* (1988), pp. 147–151.
13. HARTLEY, R., AND ZISSERMAN, A. *Multiple View Geometry in Computer Vision*, second ed. Cambridge University Press, 2004.
14. HOPPE, H., DEROSE, T., DUCHAMP, T., McDONALD, J., AND STUETZLE, W. Surface reconstruction from unorganized points. *SIGGRAPH Comput. Graph.* 26 (July 1992), 71–78.
15. HU, S., QIAO, J., ZHANG, A., AND HUANG, Q. 3d reconstruction from image sequence taken with a handheld camera. In *International Society for Photogrammetry and Remote Sensing, Congress Beijing 2008, Proceedings of Commission IV* (2008), pp. 559–562. URL: http://www.isprs.org/proceedings/XXXVII/congress/4_pdf/99.pdf.
16. HUA, S., AND LIU, T. Realistic 3d reconstruction from two uncalibrated views. *International Journal of Computer Science and Network Security* 7, 6 (2007), 178–183.
17. KAZHDAN, M., BOLITHO, M., AND HOPPE, H. Poisson surface reconstruction. In *Proceedings of the fourth Eurographics symposium on Geometry processing* (Aire-la-Ville, Switzerland, Switzerland, 2006), SGP '06, Eurographics Association, pp. 61–70.
18. LOWE, D. G. Object recognition from local scale-invariant features. In *Proceedings of the International Conference on Computer Vision-Volume 2 - Volume 2* (Washington, DC, USA, 1999), ICCV '99, IEEE Computer Society, pp. 1150–.
19. MARTINEC, D., AND PAJDLA, T. 3d reconstruction by gluing pair-wise euclidean reconstructions, or "how to achieve a good reconstruction from bad images". In *Proceedings of*

- the Third International Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT'06)* (Washington, DC, USA, 2006), 3DPVT '06, IEEE Computer Society, pp. 25–32. URL: <http://cmp.felk.cvut.cz/~martidl/articles/Martinec-3DPVT2006.pdf>.
20. MATUSIK, W., BUEHLER, C., RASKAR, R., GORTLER, S. J., AND MCMILLAN, L. Image-based visual hulls. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques* (New York, NY, USA, 2000), SIGGRAPH '00, ACM Press/Addison-Wesley Publishing Co., pp. 369–374.
 21. MUYUN, W., AND MINGYI, H. Image feature detection and matching based on susan method. In *Proceedings of the First International Conference on Innovative Computing, Information and Control - Volume 1* (Washington, DC, USA, 2006), ICICIC '06, IEEE Computer Society, pp. 322–325.
 22. SCHAFFALITZKY, F., AND ZISSERMAN, A. Multi-view matching for unordered image sets, or "how do i organize my holiday snaps?". In *Proceedings of the 7th European Conference on Computer Vision-Part I* (London, UK, UK, 2002), ECCV '02, Springer-Verlag, pp. 414–431.
 23. SCHAFFALITZKY, F., AND ZISSERMAN, A. Automated location matching in movies. *Comput. Vis. Image Underst.* 92 (November 2003), 236–264.
 24. SNAVELY, N., SEITZ, S. M., AND SZELISKI, R. Photo tourism: exploring photo collections in 3d. *ACM Trans. Graph.* 25 (July 2006), 835–846.
 25. YANG, R., AND WÜNSCHE, B. C. Life-sketch: a framework for sketch-based modelling and animation of 3d objects. In *Proceedings of the Eleventh Australasian Conference on User Interface - Volume 106* (2010), AUIIC '10, Australian Computer Society, Inc., pp. 61–70. URL: http://www.cs.auckland.ac.nz/~burkhard/Publications/AUIIC2010_YangWuensche.pdf.
 26. ZHANG, J., BOUTIN, M., AND ALIAGA, D. G. Robust bundle adjustment for structure from motion. In *Proceedings of IEEE International Conference on Image Processing* (2006), pp. 2185–2188.
 27. ZHANG, R., TSAI, P.-S., CRYER, J. E., AND SHAH, M. Shape from shading: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.* 21 (August 1999), 690–706.
 28. ZOLLHÖFER, M., MARTINEK, M., GREINER, G., STAMMINGER, M., AND SÜßMUTH, J. Automatic reconstruction of personalized avatars from 3d face scans. *Comput. Animat. Virtual Worlds* 22 (April 2011), 195–202.