A Robust System for High-Quality Reconstruction of 3D Objects from Photographs

Hoang M. Nguyen, Burkhard C. Wünsche, Patrice Delmas, Christof Lutteroth, Wannes van der Mark

Department of Computer Science, University of Auckland
justin.nguyen@auckland.ac.nz, burkhard@cs.auckland.ac.nz, p.delmas@cs.auckland.ac.nz, lutteroth@cs.auckland.ac.nz, w.vandermark@auckland.ac.nz

Abstract. Image-based modeling is rapidly increasing in popularity since cameras are very affordable, widely available, and have a wide image acquisition range suitable for objects of vastly different size. In this paper we describe a novel image-based modeling system, which produces high-quality 3D content automatically from a collection of unconstrained and uncalibrated 2D images. The system estimates camera parameters and a 3D scene geometry using *Structure-from-Motion (SfM)* and *Bundle Adjustment* techniques. The point cloud density of 3D scene components is enhanced by exploiting silhouette information of the scene. This hybrid approach dramatically improves the reconstruction of objects with few visual features. A high quality texture is created by parameterizing the reconstructed objects using a segmentation and charting approach, which also works for objects which are not homeomorphic to a sphere. The resulting parameter space contains one chart for each surface segment. A texture map is created by back projecting the best fitting input images onto each surface segment, and smoothly fusing them together over the corresponding chart by using graph-cut techniques. Our evaluation shows that our system is capable of reconstructing a wide range of objects in both indoor and outdoor environments.

1 INTRODUCTION

A key task in mobile robotics is the exploration and mapping of an unknown environment using the robot's sensors. SLAM algorithms can create a map in real time using different sensors. While the resulting map is suitable for navigation, it usually does not contain a high quality reconstruction of the surrounding 3D scene, e.g., for use in virtual environments, simulations, and urban design.

High quality reconstructions can be achieved using image input and conventional modeling systems such as Maya, Lightwave, 3D Max or Blender. However, the process is time-consuming, requires artistic skills, and involves considerable training and experience in order to master the modeling software. The introduction of specialized hardware has simplified the creation of models from real physical objects. Laser scanners can create highly accurate 3D models, but are expensive and

have a limited range and resolution. RGBD sensors, such as the Kinect, have been successfully used for creating large scale reconstructions. In 2011 the Kinect-Fusion algorithm was presented, which uses the Kinect depth data to reconstruct a 3D scene using the Kinect sensor like a handheld laser scanner [Newcombe et al. 2011]. Since then a wide variety of new applications have been proposed such as complete 3D mappings of environments [Henry et al. 2012]. The Kinect is very affordable, but has a very limited operating range (0.8 - 3.5m), a limited resolution and field-of-view, and it is sensitive to environmental conditions [Oliver et al. 2012]. Reconstruction 3D scenes from optical sensor data has considerable advantages such as the low price of cameras, the ability to capture objects of vastly different size, and the ability to capture highly detailed color and texture information. Furthermore optical sensors are very light weight and have a low energy consumption, which makes them ideal for mobile robots, such as small Unmanned Aerial Vehicles (UAVs).

This paper proposes a novel system that employs a hybrid multi-view image-based modeling approach coupled with a surface parameterization technique as well as surface and texture reconstruction for automatically creating a high quality reconstruction of 3D objects using uncalibrated and unconstrained images acquired using consumer-level cameras. In contrast to previous works we combine both correspondence-based and silhouette-base reconstruction techniques, which improves reconstruction results for featureless objects and objects with concave regions. These classes of objects often pose great difficulty for algorithms using only a single approach. As the result, our solution is able to produce satisfactory results with higher resolution for a much larger class of objects.
The system performs 3D reconstruction using the following steps:

1. Camera parameter estimation and scene geometry generation
2. Increase the density of the obtained point cloud by exploiting object's silhouette information
3. 3D surface reconstruction
4. Surface parameterisation and texture reconstruction

The remainder of this paper is structured as follows. In section 2, we review related work in the field of image-based modeling. Section 3 presents the design of our solution. Results are discussed in section 4. In section 5 we conclude the paper and discuss directions for future research.

2 PREVIOUS WORKS

3D image-based reconstruction algorithms can be classified and categorized based on the visual cues used to perform reconstruction, e.g., silhouettes, texture, shading or correspondence. Amongst them, shape-from-silhouette and shape-from-correspondence have proven to be the most well-known and successful visual

cues. Classes of reconstruction methods exploiting these visual cues can offer a high degree of robustness due to their invariance to illumination changes [Hernandez et al. 2008].

Shape-from-silhouette algorithms obtain the 3D structure of an object by establishing an approximate maximal surface, known as the visual hull, which progressively encloses the actual object. Shape from silhouette-based methods can produce surprisingly good results with a relatively small number of views, but have problems with complex object geometries, such as concave regions [Grauman et al. 2003; Matusik et al. 2000; Nguyen et al. 2012]. Most techniques extract silhouette contours [Baumgart 1974] and then derive a 3D geometry from them, e.g. by computing the intersection of silhouette cones [Martin et al. 1983]. Efficiency can be improved by using an octree representation of the visual hull [Chien et al. 1984]. [Grauman et al. 2003] use a Bayesian approach to compensate for errors introduced as the result of false segmentation.

The literature in image-based modelling describes several complete systems, but only for a limited range of applications. [Fruh et al. 2003] create textured 3D models of an entire city by using a combination of aerial imagery, ground color, and LIDAR scans, which makes the technique unpractical for consumer applications. Xiao *et al.* [Xiao et al. 2008] presented a semi-automatic image-based approach to reconstruct 3D façade models from a sequence of photographs. Quan et al. present a technique for modeling plants [Quan et al. 2006]. The algorithm requires manual interaction and makes assumptions about the geometry of the reconstructed object (e.g. leaves).

3 DESIGN

3.1 Algorithm Overview

In order to recover the scene geometry, our system automatically detects and extracts points of interest such as corners (edges with gradients in multiple directions) in the input images. The points are matched across views and changes of their relative position across multiple images are used to estimate camera parameters and 3D coordinates of the matched points using a Structure from Motion technique. The method requires that input images partially overlap.

Feature matching is achieved using an incremental approach starting with a pair or images having a large number of matches, but also a large baseline. This is to ensure that the 3D coordinates of observed points are well-conditioned. The remaining images are added one at a time ordered by the number of matches [Cheng et al. 2011; Snavely et al. 2006]. The Bundle Adjustment technique is subsequently applied to refine and improve the solution.

The density of the obtained scene geometry is enhanced by exploiting the silhouette information in the input images. The end result of this stage is a dense point cloud of the scene to be reconstructed. A 3D surface mesh is obtained by interpolating the 3D point cloud. The surface is then parameterized and a texture map is obtained by back projecting the input images and fusing them together using graph-cut techniques. Fig. 1 depicts several stages of the reconstruction process.

3.2 Camera Parameter Estimation

The objective of this stage is to recover the intrinsic and extrinsic parameters of each view. This is accomplished in two steps: First, salient features are extracted and matched across views. Second, the camera parameters are estimated using Structure-from-Motion and Bundle Adjustment techniques. In our system we use the SIFT feature detector [Lowe 1999; Lowe 2004].

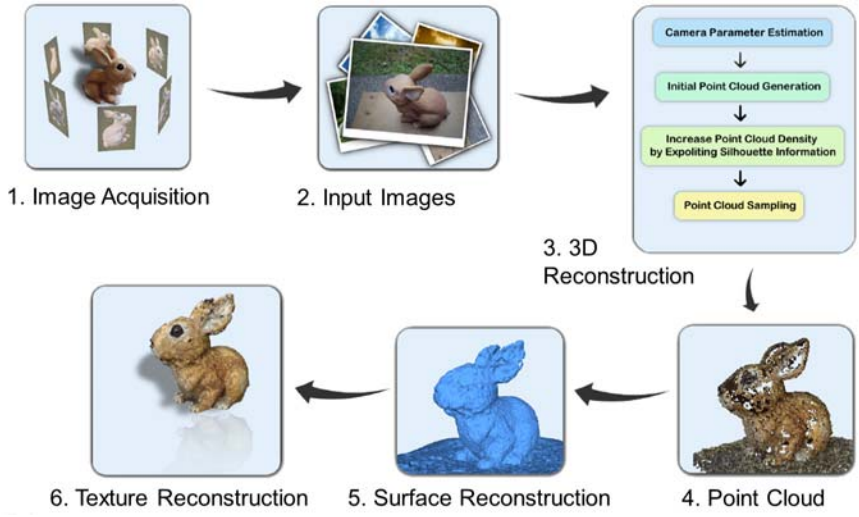

Fig. 1 Stages of the reconstruction process.

Once features have been detected and extracted from the input images, they are matched in order to find pairwise correspondences between them. This is achieved by using a distance metric to compute the similarity of each feature of a candidate image with features of another image. A small distance signifies that the two key points are close and thus similar. However, a small distance does not necessarily mean that the points represent the same feature. For instance, the corners of windows of a building look similar regardless of whether two photos show the same or different parts of the building. In order to accurately match a key point in the candidate image, we identify the closest and second closest key point in the reference image using a nearest neighbor search strategy. If their ratio is below a given

threshold, the key point and the closest matched key point are accepted as correspondences, otherwise the match is rejected [Lowe 1999; Lowe 2004].

At this stage, we have a set of potentially matching image pairs, and for each pair, a set of individual feature correspondences. For each pair of matching images, we compute the Fundamental Matrix using the RANSAC algorithm. Erroneous matches are eliminated by enforcing epipolar constraints. Scene geometry and the motion information of the camera are estimated using the Structure-from-Motion technique [Cheng et al. 2011; Snavely et al. 2006; Szeliski 2006], and are further refined using Bundle Adjustment.

3.2 Scene Geometry Enhancement

At this stage, we have successfully acquired both the camera parameters and the scene geometry. Due to the sparseness of the scene geometry, the surface and texture reconstruction frequently produce artefacts. Most previous works approached this problem by constraining the permissible object types or requiring manual hints for the reconstruction process. However, these requirements breach our goal of creating an easy-to-use system capable of reconstructing any type of object where shape and texture properties are correctly captured by the input photos.

We improve the reconstruction results by exploiting the silhouette information to further enrich the density of the point cloud: First, the silhouette information in each image is extracted using the Marching Squares algorithm [Lorensen et al. 1995], which produces a sequence of all contour pixels. To construct a visual hull representation of the scene using an immense silhouette contour point set will inevitably increase computational costs. In order to avoid this, the silhouette contour data is converted into a 2D mesh using a Delaunay triangulation, and the mesh is simplified using a mesh decimation algorithm [Melax 1998]. This effectively reduces the number of silhouette
contour points. A point cloud representing the visual hull of the scene is obtained using a technique presented by [Matusik et al. 2000].

3.3 Surface Reconstruction

At this stage we have successfully obtained a quasi-dense 3D point cloud, which in the next step needs to be approximated by a smooth closed surface (without holes) that represents the underlying 3D model from which the point cloud was obtained. We tested several surface reconstruction techniques including the power crust algorithm [Amenta et al. 2001], α -shapes [Edelsbrunner 1995], and the ball-pivoting algorithm [Bernardini et al. 1999]. We decided to employ the Poisson Surface Reconstruction algorithm [Kazhdan et al. 2006], since it produces a closed surface and works well for noisy data. In contrast to many other implicit surface fitting methods, which often segment the data into regions for local fitting and

then combine these local approximations using blending functions, Poisson surface reconstruction processes all the sample points at once, without resorting to spatial segmentation or blending [Kazhdan et al. 2006].

3.3 Texture Reconstruction

A high-resolution texture for the reconstructed 3D object is obtained by parameterizing the 2D mesh and computing a texture map.

a) Surface Parameterization: We tested surface parameterization algorithms provided by existing libraries and tools, such as Blender. We found that they required manual hints, only worked for objects homeomorphic to a sphere, or created a surface parameterization using many disconnected patches. The latter result is undesirable since it creates visible seams in the reconstructed texture, and since it makes post-processing steps, such as mesh reduction, more difficult.

In order to use the resulting 3D models in a large variety of applications and professional production pipelines, we need a texture map which consists of a small number of patches, which ideally correspond to geometric features (which can be maintained in a post-processing step such as mesh reduction). The Feature-based Surface Parameterization technique by Zhang et al. fulfils these criteria [Zhang et al. 2005]. The algorithm consists of three stages:

1. Genus reduction: In order to identify non-zero genus surfaces, a surface-based Reeb graph [Reeb 1946] induced by the average geodesic distance [Hilaga et al. 2001] is constructed. Cycles in the graph signify the existence of handles/holes in the surface, i.e., the surface is not homeomorphic to a sphere. Examples are donut and teacup shaped objects. The genus of the surface is reduced by cutting the surface along the cycles of the graph. The process is repeated until there are no more cycles.

2. Feature identification: Tips of surface protrusions are identified as leaves of the Reeb graph. The features are separated from the rest of the surface by constructing a closed curve.

3. Patch creation: The previous two steps segment the surface into patches which are homeomorphic to a disk. Patches are “unwrapped” using discrete conformal mappings [Eck et al. 1995]. The algorithm first positions the texture coordinates of the boundary vertices, and then finds the texture coordinates of the interior vertices by solving a closed form system. Distortions are reduced by using a post-processing step, which optimizes the position of interior vertices’ texture coordinates by first computing an initial harmonic parameterization [Floater 1997] and then applying a patch optimization technique [Sander et al. 2002].

The image on the left of Fig. 2 illustrates the resulting parameterization of our Rooster model. Each disk in the 2D texture map corresponds to a surface segment of the 3d model.

b) Texture Generation: The texture map for the parameterized surface is computed in three steps:

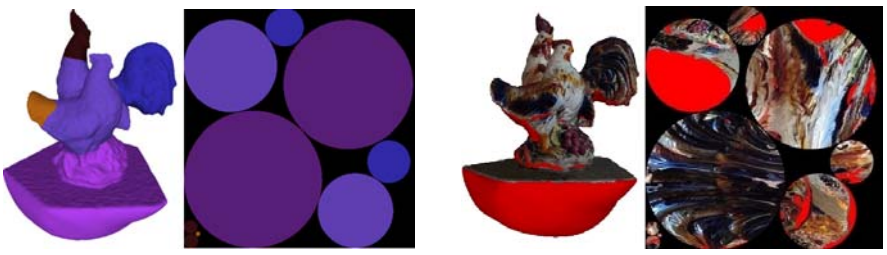

Fig. 2 The Rooster model: The segmented 3D model and the corresponding texture atlas (left) and the reconstructed texture obtained by projecting and fusing input photographs (right).

1. Identify regions of input images: The objective of this step is to compute for each patch of the texture map (the disks in the second image from the left in Fig. 2) pixel colors, which accurately represent the surface colors of the 3D object at the corresponding points. This is achieved by projecting the corresponding surface patch, one triangle at a time, onto all input images where it is visible. We call the resulting section of the input image the back projection map and we call the resulting mapping between surface triangles and input image regions the back-projection mapping. The projection is only performed if the angle between a triangle's normal and the ray shooting from the triangle's centroid to the estimated camera position of the input image is larger than 90^{O}.

2. **Texture map computation:** The image regions defined by the back projection map define the color information for the corresponding patch of the texture map. Using back projection mapping and the surface parameterization we can compute for each triangle of the surface mesh a mapping from the input image to the texture's parameter space. The algorithm is repeated for all patches of the reconstructed surface texture region and yields a set of overlapping textures covering the object.

3. Minimize seams between overlapping textures: Seams between overlapping textures are minimized by using a graph cut algorithm [Kwatra et al. 2003]. We investigated different parameters settings for image fusion applications and found that Kwatra et al.'s cost function (gradient weighted color distance) in combination with the RGB color space and the L_2 norm works well for most applications [Clark et al. 2012].

The rightmost image in Fig. 2 shows the texture map obtained by back projection surface patches onto the input images and the resulting textured 3D model. Regions where no texture information was recovered are indicated in red. A typical reason is that users forget to make photos of the underside of the imaged object. In this case the Poisson surface reconstruction will still create a smooth surface interpolating the gaps in the point cloud, but no texture is available since that part of the surface is not shown on any input image. Fig. 3 illustrates the level of detail obtainable with our texture reconstruction process.

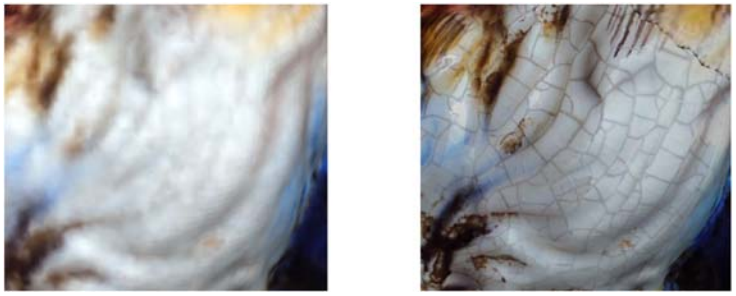

Fig. 3 Texture reconstruction by computing vertex colors and interpolating them (left) and the texture obtained using our approach (right). Note that both images show the neck section of the rooster in Fig. 2. The cracks in the image on the right reflect accurately the appearance of the object's material.

4 RESULTS

We tested our image-based modeling system using more than 40 data sets of both indoor and outdoor scenes, and of objects of different scale. Our system produces qualitatively good results for both uniformly colored and feature-poor objects, and for objects with concave regions and moderately complex geometries. The size of our test datasets varied from as few as 6 images to hundreds of images. All input images were acquired with a simple consumer-level handheld camera, including a Smartphone camera. The average computation time varies between 12 minutes to 10 hours. Our system fails for objects which have viewpoint dependent surface appearance, e.g., refractive and reflective materials within complex environments. The following paragraphs present three examples of our results.

4.1 Horse Model

The dataset consists of 37 images of a wooden horse model. The images were acquired outdoors on a sunny day and have a resolution of 2592 x 1944 pixels. Three of the images are shown on the left of Fig. 4. The original object has a very smooth, reflective and shiny surface with few distinctive visual features. The resulting reconstructed model, shown on the right of Fig. 4, is of excellent quality and bears a high resemblance to the original object. The resulting model consists

of 329,275 polygons and requires approximately 5 hours and 12 minutes on an Intel Quad Core i7 with 6GB RAM.

Fig. 4 Three out of 37 input images of the horse model data set (left) and the resulting reconstructed 3D model (right).

4.2 Miniature House Model

This dataset consists of 27 images of a replica of the famous house in Alfred Hitchcock's movie "Psycho". The images have a resolution of 2592 x 1944 pixels and were acquired with a consumer-level SONY DSCW18 camera under complex lighting condition (multiple spotlights and diffuse lights). The model's surface has a complex shape with many small features and holes.

The resulting reconstructed object (right hand side of Fig. 5) consists of 208,186 polygons and has an acceptable visual quality. The detailed fence-like structure on top of the roof and the tree leaves could not be accurately reconstructed since they were too blurry in the input images. Hence neither the shape-from correspondence approach, nor the shape-from-silhouette approach could create a sufficiently high number of points for capturing the 3D geometry. The computation time of this data set was 4 hours 21 minutes on an Intel Quad Core i7 with 6GB RAM

Fig. 5 One of 27 input images of a miniature house model (left) and the resulting reconstructed 3D model (right).

4.3 Elephant Model

The elephant model consists of 21 images as illustrated on the left of Fig. 6. The images have a resolution of 2592 x 1944 pixels and were acquired with a consumer-level SONY DSCW180 camera in an indoor environment with relatively low light setting. The object has a complex surface geometry with many bumps and wrinkles, but few distinctive textural features. The resulting 3D reconstruction, shown on the right of Fig. 6, has 198,857 faces and is of very good quality. The texture and surface geometry of the object contain surprisingly accurate surface details. This example illustrates that our system performs well for objects with dark, rough surfaces and under relatively poor lighting conditions with large illumination variations and shadowing. The reconstruction process took almost 3 hours to complete on an Intel Quad Core i7 with 6GB RAM.

Fig. 6 Two out of 21 input images of the elephant model data set (top) and the resulting 3D reconstruction (bottom).

5 CONCLUSIONS AND FUTURE WORK

We have described a novel image-based modelling system creating high quality 3D models fully automatically from a moderate number (20-40) of camera images. Input images are unconstrained and uncalibrated, which makes the system especially useful for low-cost and miniature mobile robots. In contrast to laser scanners our system also works for shiny and dark objects. The system still has some drawbacks which need to be addressed in future research. Missing regions in the

texture map occur if the input images do not cover the entire object. We are currently working on texture inpainting techniques to fill these regions [Bertalmio et al. 2000; Perez et al. 2003].

6 ACKNOWLEDGEMENTS

We would like to thank Prof. Eugene Zhang from the Oregon State University for providing us with code for his Feature-based Surface Parameterization technique [Zhang et al. 2005] and assisting with integrating it in our system.

7 REFERENCES

[Amenta et al. 2001]
Amenta N, Choi S, Kolluri RK (2001) The power crust. In Proceedings of the sixth ACM symposium on Solid modeling and applications (SMA '01), ACM Press, pp 249–266

[Baumgart 1974]
Baumgart BG (1974) Geometric modeling for computer vision. Doctoral Dissertation, Stanford University

[Bernardini et al. 1999]
Bernardini F, Mittleman J, Rushmeier H, Silva C, Taubin G (1999) The ball-pivoting algo-rithm for surface reconstruction. IEEE Transactions on Visualization and Computer Graphics, 5(4):349–359

[Bertalmio et al. 2000]
Bertalmio M, Sapiro G, Caselles V, Ballester C (2000) Image inpainting. In Proceedings of the 27th annual conference on Computer graphics and interactive techniques (SIGGRAPH '00), pp 417–424

[Cheng et al. 2011]
Cheng W, Ooi WT, Mondet S, Grigoras R, Morin G. (2011) Modeling progressive mesh streaming: Does data dependency matter. ACM Transaction on Multimedia Computing, pp 1–24

[Chien et al. 1984]
Chien CH and Aggarwal JK (1984) A volume surface octree representation. In Seventh Inter-national Conference on Pattern Recognition, Montreal, Canada, pp 817–820

[Clark et al. 2012]
Clark XB, Finlay JG, Wilson AJ, Milburn KLJ, Nguyen HM, Lutteroth C, Wunsche BC (2012) An investigation into graph cut parameter optimisation for image-fusion applications. IVCNZ '12 Proceedings of the 27th Conference on Image and Vision Computing New Zealand, pp 480–485

[Eck et al. 1995]
Eck M, DeRose M, Duchamp T, Hoppe H, Lounsbery M, Stuetzle W (1995) Multiresolution analysis of arbitrary meshes. In Proceedings of the 22nd annual conference on Computer graphics and interactive techniques (SIGGRAPH '95), pp 173–182

[Edelsbrunner 1995]
Edelsbrunner H (1995) Smooth surfaces for multi-scale shape representation. In Proceedings of the 15th Conference on Foundations of Software Technology and Theoretical Computer Science, pp 391–412

[Floater 1997]
Floater MS (1997) Parametrization and smooth approximation of surface triangulations. Computer Aided Geometric Design, 14(3):231–250.

[Franco et 2006]
Franco JS, M. Lapierre, and E. Boyer. Visual shapes of silhouette sets. In 3D Data Processing, Visualization and Transmission, pp 397–404

[Fruh et al. 2003]
Fruh C, Zakhor A (2003) Constructing 3D city models by merging ground-based and airborne views. Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, 2:562–569

[Grauman et al. 2003]
Grauman K, Shakhnarovich G, Darrell T (2003) A bayesian approach to image-based visual hull reconstruction. In IEEE International Conference on Computer Vision and Pattern Recognition, 1:187–194

[Henry et al. 2012]
Henry P, Krainin M, Herbst E, Ren X, Fox D (2012) RGB-D mapping: Using kinect-style depth cameras for dense 3d modeling of indoor environments. Int. J. Rob. Res, 31(5):647–663

[Hernandez et al. 2008]
Hernandez C, Vogiatzis G, Cipolla R (2008) Multi-view photometric stereo. In IEEE Transaction on Pattern Recognition and Machine Intelligence, 30:548–554

[Hilaga et al. 2001] Hilaga M, Shinagawa Y, Komura T, Kunii TL (2001) Topology matching for fully automatic similarity estimation of 3D shapes. Computer Graphics Proceedings, Annual Conference Series (SIGGRAPH 2001), pp 203–212

[Kazhdan et al. 2006]
Kazhdan M, Bolitho M, Hoppe H (2006) Poisson surface reconstruction. In Proceedings of the fourth Eurographics symposium on Geometry processing (Aire-la-Ville, Switzerland, Switzerland, 2006), pp 61–70

[Kwatra et al. 2003]
Kwatra V, Schödl A, Essa I, Turk G, Bobick A (2003) Graphcut textures: image and video synthesis using graph cuts. ACM Trans. Graph., 22(3):277–286.

[Lorensen et al. 1995]
Lorensen WE (1995) Marching through the visible man. In Proceedings of the 6th conference on Visualization ’95, pp 368–373

[Lowe 1999]
Lowe DG (1999) Object recognition from local scale-invariant features. In International Con-ference on Computer Vision, 2:1150–1157

[Lowe 2004]
Lowe DG(2004) Distinctive image features from scale-invariant keypoints. In International Journal of Computer Vision, 60:91–110

[Martin et al.1983]
Martin W and Aggarwal JK (1983) Volumetric descriptions of objects from multiple views. In IEEE Transactions on Pattern Analysis and Machine Intelligence, 5(2):150–158

[Matusik et al. 2000]
Matusik W, Buehler C, Raskar C, Gortler SJ, McMillan L (2000) Image-based visual hulls. In Proceedings of the 27th annual conference on Computer graphics and interactive techniques (SIGGRAPH ’00), pp 369–374

[Melax 1998]
Melax S (1998) Simple, fast, and effective polygon reduction algorithm. Game Developer Magazine, pp 44–49

[Newcombe et al. 2011]
Newcombe RA, Izadi S, Hilliges O, Molyneaux D, Kim D, Davison AJ, Kohli P, Shotton J, Hodges S, Fitzgibbon A (2011) KinectFusion: Real-time dense surface mapping and tracking. In Proceedings of the 10th IEEE International Symposium on Mixed and Augmented Reality (ISMAR '11), pp 127–136

[Nguyen et al. 2012]
Nguyen HM, Wunsche BC, Delmas P, Lutteroth C (2011) Realistic 3d scene reconstruc-tion from unconstrained and uncalibrated images. In Proceedings of GRAPP 2011, Algarve, Portugal, 31:67–75

[Oliver et al. 2012]
Oliver A, Kang S, Wünsche BC, MacDonald B (2012) Using the kinect as a navigation sensor for mobile robotics. In Proceedings of IVCNZ 2012, pp 509–514

[Perez et al. 2003]
Perez P, Gangnet M, Blake A (2003) Poisson image editing. ACM Trans. Graph., 22(3):313–318

[Quan et al. 2006]
Quan L, Tan P, Zeng G, Yuan L, Wang J, Kang SB (2006) Image-based plant modeling. In ACM Transactions on Graphics, 25(3):599–604

[Reeb 1946]
Reeb G (1946) Sur les points singuliers dune forme de pfaff completement integrable ou diune fonction numerique [on the singular points of a completely integrable pfaff form or of a numerical function. Comptes Randus Acad. Sciences Paris 222, pp 847–849

[Sander et al. 2002]
Sander PV, Gortler SJ, Snyder J, Hoppe H (2002) Signal-specialized parameterization. Proceedings of the 13th Eurographics Workshop on Rendering, pp 87–100

[Snavely et al. 2006]
N. Snavely, S. Seitz, and R. Szeliski. Photo tourism: Exploring photo collections in 3D. In ACM Transactions on Graphics, 25(3):835–846

[Szeliski 2006]
Szeliski R. Image alignment and stitching. A tutorial in Computer Graphics and Vision

[Xiao et al. 2008]
Xiao J, Fang T, Tan P, Zhao P, Ofek E, Quan L (2008) Image-based façade modeling. In ACM Transactions on Graphics, 27(5):26–34

[Zhang et al. 2005]
Zhang E, Mischaikow K, Turk G (2005) Feature-based surface parameterization and texture mapping. ACM Transactions on Graphics (TOG), 24(1):1–27