Identifying Low Confidence Mesh Regions: Uncertainty Measures and Segmentation

Hoang Minh Nguyen
The University of Auckland
Auckland, New Zealand
justin.nguyen@auckland.ac.nz

Burkhard Wünsche
The University of Auckland
Auckland, New Zealand
burkhard@cs.auckland.ac.nz

Patrice Delmas
The University of Auckland
Auckland, New Zealand
p.delmas@cs.auckland.ac.nz

Christof Lutteroth
The University of Auckland
Auckland, New Zealand
lutteroth@cs.auckland.ac.nz

ABSTRACT

3D digital models have become an important part of diverse applications ranging from computer games, virtual reality, architectural design to visual impact studies. One common method to create 3D models is to create a point cloud using laser scanners, structured lighting sensors, or image-based modelling techniques, and then construct a 3D mesh, and texture-map it using photographs of the observed scene. Attributed to the inherent properties of general 3D scenes such as occluded or inaccessible parts, reflective surfaces, lighting conditions or poor-quality inputs, 3D models produced by these approaches often exhibit unsatisfactory and erroneous mesh regions. In many cases, it is desirable to identify and extract such regions so that they can be constructed or corrected through other means. While much effort has been invested into the problem of 3D reconstructions, the task of evaluating existing models and preparing them for subsequent enhancement processes has been largely neglected. In this paper, we present a novel method for automatically detecting and segmenting mesh regions with low confidence in their correctness. The confidence estimation is achieved by exploiting and integrating various uncertainty measures such as geometric distances, normal variations and texture discrepancies. Low-confidence mesh regions are isolated and removed in such a way that the extracted region's boundary is as simple as possible in order to facilitate subsequent automatic or manual improvement of these regions. Segmentation is achieved by minimising an energy function that takes the genus and boundary length and smoothness of the extracted regions into account.

IVCNZ '14, November 19 - 21 2014, Hamilton, New Zealand
Copyright 2014 ACM 978-1-4503-3184-5/14/11$15.00
http://dx.doi.org/10.1145/2683405.2683409

Categories and Subject Descriptors

D.3.2 [**3D Reconstruction**]: Mesh manipulation—*mesh classification, mesh segmentation*

Keywords

3D Reconstruction, Mesh Classification, Uncertainty Measure

1. INTRODUCTION

There is an increasing amount of applications that necessitate high quality 3D representations, e.g. for arts, commerce, virtual heritage, training, education, computer games, virtual environments, documentation, exchanging information, and social networking applications. The introduction of specialised hardware, such as laser and structured lighting sensors [5], and the emergence of various sophisticated image-based modelling techniques [2, 4, 6, 7, 11] has simplified the creation of virtual 3D models from real physical objects.

Although these approaches enable the reconstruction of plausible and comprehensive 3D models, the reconstruction quality often hinges on various additional factors such as lighting and surface materials when a laser scanner is used or insufficient input images and unfavourable lighting conditions in the case of image-based methods. Violation of these requirements often results in degraded regions in the resultant 3D models. While much interest has been invested in improving the status-quo of general 3D reconstruction techniques, aiming to further refine existing techniques, not much attention has been spent on identifying and correcting mesh regions which have a high likelihood of being incorrect. We were unable to identify any previous research providing a confidence or uncertainty measure for 3D reconstructions. Several authors have tried to identify problematic steps in the reconstruction process and to address them during the reconstruction process. For example, Shan *et al.* [10] identify sparse regions in the point cloud due to occlusions and incorporate that information in the Poisson surface reconstruction by identifying contours and estimating depth values for the occluded regions.

We propose a novel method for classifying and extracting uncertain mesh regions in a reconstructed 3D model. Our method employs and fuses various uncertainty measures for

the detection of regions with low-confidence. Once identified, these regions are isolated and removed through a boundary-energy-minimising process, which defines a mesh region that contains low-confidence mesh sections, and which has the lowest possible genus and boundary length and the highest possible boundary smoothness. Determining a mesh section with a simple shape enclosing the low-confidence mesh regions is important for subsequent post-processing steps improving these regions, i.e. by sketch-based modelling [1]. The need for simple boundaries of low-confidence regions is demonstrated by that fact, that many surface interpolation techniques are energy-minimising and performed poorly in our experiments when applying them to complex contours.

The remainder of this paper is organised as follows. In section 2, we present our algorithm for identifying low-confidence mesh regions. Section 3 discusses our mesh segmentation approach. Results are shown in section 4. Section 5 concludes this paper and gives an outlook on future work.

2. MESH CLASSIFICATION

The mesh confidence classification is accomplished through the integration of three measurements: geometric distance, normal deviation, and texture discrepancy. In the following sections, a detailed description of each of these measurements will be presented and discussed.

2.1 Classification by Geometric Distance

Meshes are frequently compared using geometric distances. A popular measure to compare two polygon meshes A and B is the two-sided Hausdorff distance which is defined as:

$$D(A,B) = max\Big\{h(A,B),\, h(B,A)\Big\} \tag{1}$$

$$h(A,B) = \max_{a\in A}\left\{\min_{b\in B}\left\{d(a,b)\right\}\right\} \tag{2}$$

where $d(a,b)$ denotes the distance between two faces: a and b. We assume that we are dealing with a mesh obtained by creating a 3D point cloud, which is subsequently interpolated, e.g. using Poison Surface reconstruction. In many application scenarios, the point cloud has sparse areas and the surface reconstruction technique introduces errors, e.g. by using a smoothing process.

We hence propose as a confidence measure of the reconstruction quality the geometric distance $\sigma_{distance}(F)$ between the point cloud and each face F of the resulting polygon mesh. The measure will produce a large value (uncertainty) in areas where the polygon cloud is sparse (since many mesh polygons do not have any nearby points). The distance measure will also be relatively large in areas where the surface reconstruction method performs a large amount of smoothing, but it will be small where the point cloud is dense and the surface interpolates the point cloud well.

To compute the distance between a 3D point to a particular face, we employ the point-triangle distance algorithm presented in [3]. The computation is sped up using a spatial subdivision scheme.

Figure 1: 3D point clouds of a historic building using an image-based modelling technique with 21 input images.

Figure 1 depicts the point cloud of a historic building, produced by an image-based modelling technique using 21 input images showing the front and sides of the house. The back and the top side of the building were not accessible.

The uncertainty measure using *geometric distance* for the mesh produced by interpolating this point cloud is shown in figure 2. Red colours signify regions with high uncertainty, and are most visible for the roof and the back of the building, where the point cloud is very sparse. Thus, by scrutinising the geometric distances between a face and its closest 3D vertex, we can effectively identify missing regions of a given 3D structure.

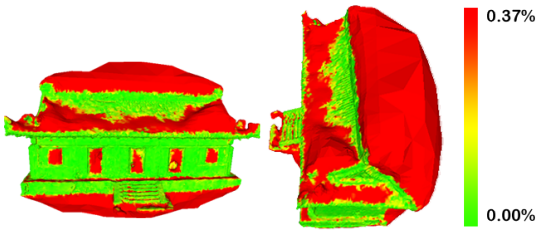

Figure 2: The 3D reconstruction of the historic building in figure 1 colour coded with our distance measure for estimating the uncertainty in the surface reconstruction accuracy. The colour scale on the right indicates the point-polygon distance as a percentage of the diagonal of the bounding box of the model. Errors of equal or greater than 0.37% are indicated in red.

2.2 Classification by Normal Direction Distribution

In some 3D acquisition methods a point cloud with vertex normals is obtained. For instance, in image-based modelling, normals are often generated by averaging viewing vectors from a 3D point to all observing camera centres. For other 3D reconstruction techniques, such as laser scanning-based 3D reconstruction, the point cloud does not have normals and surface normals are defined after a surface mesh has been created by triangulating/interpolating the point

cloud. In both cases, rapid normal direction changes are often an indicator of mesh regions deviating from the original 3D model, e.g., because complex surfaces are more difficult to reconstruct, sensor sample rates are often insufficient to capture geometric details, and because mesh errors are often associated with sudden jumps in the normal direction. For this reason, the distribution of mesh normal directions $\sigma_{normal}(F)$ for a face F can be exploited to provide an estimate for the accuracy of the reconstructed surface.

The task of classifying and segmenting low-confidence mesh regions using normal direction is accomplished as follows. The entire scene is first partitioned into a regular 3D Cartesian grid. In our experiments we used 100 grid cells in each coordinate direction. The points of the 3D point cloud and the polygons of the 3D mesh are assigned to the grid cell enclosing them.

In order to estimate the uncertainty of a mesh polygon we observe that natural objects often exhibit smooth surfaces. Likewise, surfaces of man-made objects often contain large flat surfaces with clearly defined, often orthogonal, angles between them. In the first case normal directions are fairly regular distributed, whereas in the second case normal directions are clustered along a few dominant directions. On the other hand, normal directions of poorly reconstructed surfaces tend to exhibit a much higher degree of randomness as illustrated in figure 3.

Figure 3 displays on the left the mesh triangle normals of our *Saint George Church* model. The enlargement shows that normal directions vary greatly around the window. We can see from the image on the right that the window should have sharp corners and the reconstructed surface is indeed a poor representation of the original object.

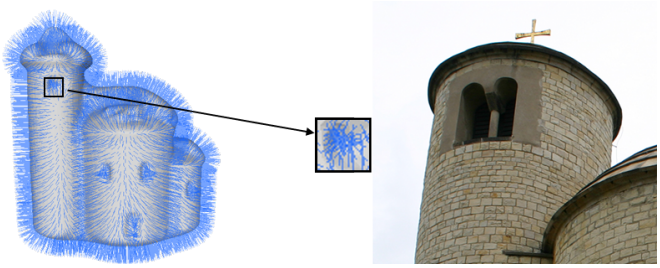

Figure 3: Normals of Saint George Model.

In order to separate smooth normal direction distributions, clustered normal direction distributions, and more random normal direction distributions, we use the following algorithm: we first approximate the angle between each normal pair by computing the length of the cross product of each normal pair. We then compute the mean value and standard deviation of the resulting values. For smoothly changing normals and for clustered normal directions there are many normal pairs with a small angle between them, whereas for randomly distributed normal directions the standard deviation tends to be larger. We hence use the standard deviation of the length of the cross products of normal pairs as an indication of the randomness of normal directions and hence as an indicator for the uncertainty in the reconstructed surface for all faces F in the corresponding grid cell.

Figure 4 displays the 3D reconstruction of the Saint George Model colour coded with our new normal measure for estimating the uncertainty in the surface reconstruction accuracy.

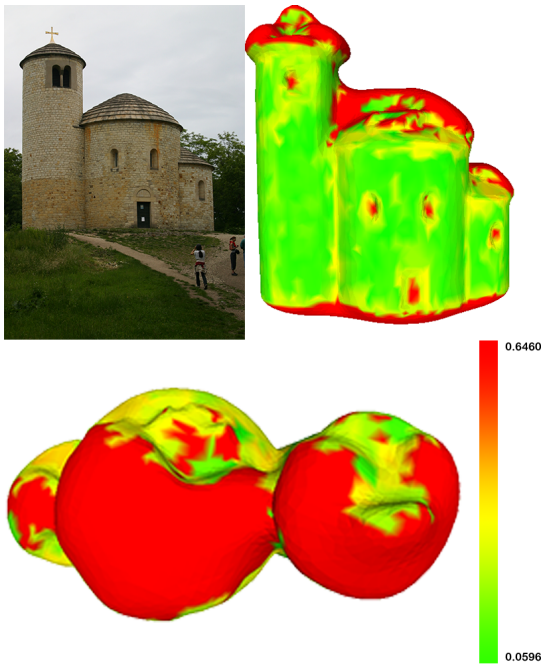

Figure 4: The 3D reconstruction of the Saint George Model colour coded with our new normal measure for estimating the uncertainty in the surface reconstruction accuracy. The colour scale on the right indicates our estimate for the randomness in normal directions.

2.3 Classification by Texture Inconsistency

Possible wrongly reconstructed mesh regions can also be identified by determining inconsistencies between input images and reconstructed textures. We assume that the surface has been parameterised and a texture has been reconstructed from a set of photographs of the 3D model [8]. The parameterisation represents a one-to-one mapping between each face of the model and a 2D triangle correspondence. An uncertainty value for each polygon of the reconstructed mesh $\sigma_{texture}(F)$ is computed as follows:

Given a face F of the reconstructed 3D model, for each input image, the associated camera parameters are used to decide if that image is within the field-of-view of the face F. First, the centre of projection is computed from the camera parameters. Next, two vectors are constructed. One goes from the centroid of face F to the centre of projection, and the other denotes the normal of the face. To determine if a given face is inside the field-of-view of the current camera, the angle subtended by those vectors is examined. If it is above a predefined threshold (empirically set to 50°) then that image is outside the field of view of the current camera and thus ignored.

For each image that passes the field-of-view test, the corresponding texture of the face F from that view is extracted

and stored in a list. To extract a patch of texture from a given image, we project the 3 vertices of F onto the image, yielding a triangle. Then we can compute an affine transformation that maps this projected triangle to a triangular region within the parameter space. By the end of this process, for each face, there will be a list of texture patches. The error value of face F is computed as follows:

$$\sigma_{texture}(F) = \begin{cases} \infty & \text{if } n = 0 \\ \alpha_i & \text{if } n = 1 \\ \frac{1}{n} \sum_{i=0}^{n-1} \sum_{j=i+1}^{n} cos(\alpha_i)\ cos(\alpha_j)\ d_{ij} & \text{if } n > 1 \end{cases} \quad (3)$$

where n denotes the number of texture patches, α_i and α_j are the angles between the face normal and view vector associated with the input image used to create the texture patches i and j, and d_{ij} represents the discrepancy between the two texture patches. The dissimilarity score d_{ij} of a pair of texture patches is obtained by comparing their differences using appearance space attributes. The following appearance space attributes are used for comparison:

- RGB color of the patches
- Gradient values in both x and y directions.
- The intensity variance
- The edge counts

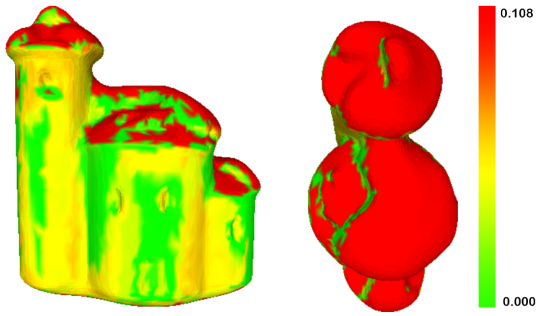

Figure 5: Classification of the Saint George model using texture inconsistency. The red colours show regions that have error scores above 0.108% of the length of the model's diagonal.

The entire information of each patch is at this stage encapsulated in a high-dimensional vector. The two vectors are then projected onto lower dimensions using PCA (8 dimensions) and compared using the norm L_2. The clear advantage of attribute space over the conventional *Sum of Squared Differences* is that the attribute space approach permits any meaningful information about the pixels and their surrounding to be embedded for comparison purposes. By reducing the dimensionality, the computation time can be kept manageable.

Figure 5 displays the 3D reconstruction of the Saint George Model colour coded with our new texture inconsistency measure for estimating the uncertainty in the surface reconstruction accuracy.

The total uncertainty is computed as weighted average of the previously introduced uncertainty measures:

$$\sigma_{total} = 5\,\sigma_{distance} + 2\,\sigma_{normal} + 1.43\,\sigma_{texture} \quad (4)$$

The weighting factors were determined by analyzing the three uncertainty measure for a range of models and visually determining the values where an uncertainty measure typically corresponds to a poorly reconstructed surface.

3. MESH SEGMENTATION

Once low-confidence regions have been successfully detected, they are extracted. In our system, the problem of extracting low-confidence regions is formulated as an energy minimisation problem, in which the low-confidence region of interest gradually expands in such a way that the new region contains no hole and its boundary is as short and smooth as possible. This is accomplished by iteratively expanding the region while minimising the following objective energy function:

$$E = \alpha l + \beta s + \left[\, len(h) * \infty \,\right] \quad (5)$$

where l denotes the length of the boundary, and s represents its smoothness. α and β designate weighting factors and are set to 0.4 and 0.6, respectively. The values were empirically determined and reflect that in cases where decisions have to be made between optimising for smoothness or length, the smoothness constraint should prevail. h signifies holes exhibited in the region. By imposing a substantial penalty for the existence of holes, we ensure that the resulting segmented region will have a genus of zero, i.e. contains no holes.

The length of a boundary is computed as the sum of all its edges' length, while its smoothness is defined as the sum of all internal angles between neighbouring edges. Figure 6 shows an example where low-confidence regions are extracted from the model using our method.

4. RESULTS

We have evaluated our method using a wide range of test objects, showing how the proposed technique can be applied to different types of 3D scenes. We also present an assessment on our method's accuracy using a ground-truth model obtained by a laser scanner.

To evaluate the effectiveness of our algorithm, we created a 3D model of an *owl* using the image-based modelling system described in [7]. A second model was obtained using a *Minolta VIVID 910 3D* Laser Scanner. The two models were aligned using an iterative closest point algorithm [9] and their differences were computed using the two-sided Hausdorff distance.

We then computed a confidence measure for the reconstruction quality of the model obtained by image-based modelling. This confidence measure was then compared with the

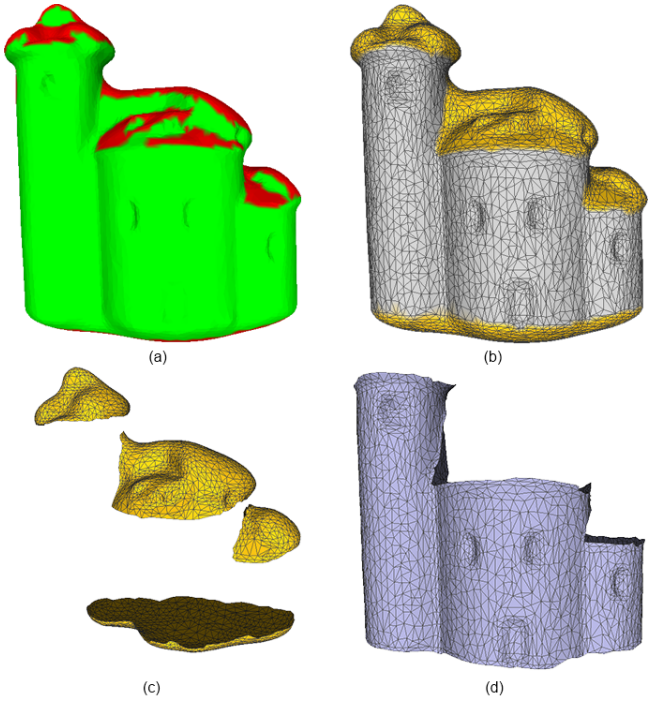

Figure 6: Segmentation of mesh regions with high uncertainty: The reconstructed 3D model with surface regions labelled red where the uncertainty measure exceeds a pre-defined threshold (a). The mesh regions in red are expanded using the algorithm described in section 3 (b) and are extracted (c). Part (d) of the figure shows the remaining mesh.

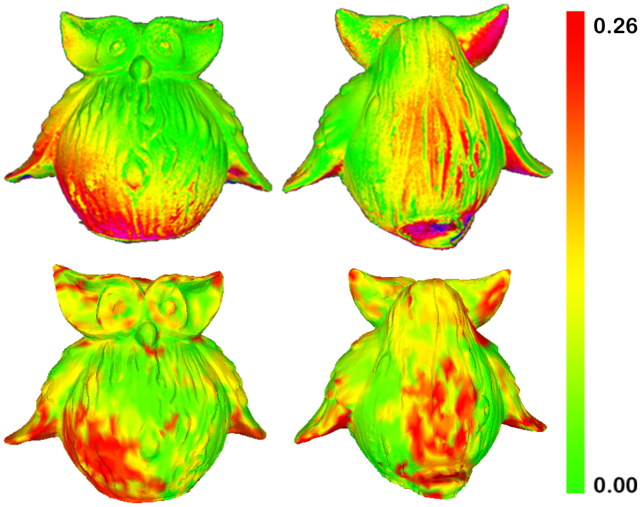

Figure 7: The top row: error-encoded model obtained by computing the two-sided Hausdorff distance between the model reconstructed using our image-based modelling technique and the laser scanner model. Bottom row: Model colour mapped with our novel mesh confidence measure. Comparing the models shows in general a good correspondence between confidence in the reconstructed model's correctness and the actual error when compared to the laser scanned model. The colour scale from green to yellow to red indicates an increasing geometric error. Red colours in this example signify regions that have error scores above 0.26% of the length of the model's diagonal

actual geometric error compared to the laser scanner-based model. Figure 7 shows the reconstructed model colour coded with the actual geometric error (top) and our new confidence measure for the reconstruction quality (bottom).

Figure 7 shows that our confidence measure is a good indicator of the actual geometric error of the reconstructed surface.

Figure 8 and 9 illustrate the effectiveness of our method through the classification of low-confidence regions of a shoe and a rooster model.

5. CONCLUSION AND FUTURE WORK

We have described a novel method for classifying and removing mesh regions with a high uncertainty of their geometric accuracy in a reconstructed 3D model. Low-confident mesh areas are identified and isolated through the integration of three separate measures: *geometric distance*, *normal direction distribution*, and *texture inconsistency*. Once successfully identified, these regions are isolated and removed through a energy-minimising process. The objective is to make certain that the resulting region has the lowest possible genus and boundary length, and the highest possible boundary smoothness.

Our results have demonstrated that our algorithm correctly identifies many poorly reconstructed mesh regions. It is possible to construct counter examples where the algorithm would fail. For example, a rough surface would always have a high value for σ_{normal} and hence at least moderately high value for σ_{total}. However, in all our tests using real 3D reconstructed models our proposed method always achieved satisfactory results.

In future work we will test our algorithm using a larger range of models obtained using different sensors and reconstruction techniques. We also would like to improve the formula for combining the three uncertainty measures in order to take into account the different priorities of these measures. For example, a rocky terrain has a high value for σ_{normal}. However, if the surface is densely sampled and $\sigma_{distance}$ is small, then the surface is likely to have a high accuracy and σ_{total} should be small.

6. REFERENCES

[1] S.-H. Bae, R. Balakrishnan, and K. Singh. Ilovesketch: As-natural-as-possible sketching system for creating 3d curve models. In *Proceedings of the 21st Annual ACM Symposium on User Interface Software and Technology*, UIST '08, pages 151–160, 2008.

[2] P. Debevec, C. Taylor, and J. Malik. Modeling and rendering architecture from photographs: A hybrid geometry and image-based approach. In *ACM Transactions on Graphics*, pages 11–20, 1996.

Figure 8: The 3D reconstruction of a shoe using image-based modelling (middle row), two input images from approximately the same view point (top row), and the mesh colour-coded with the uncertainty σ_{total} (bottom row). It can be seen that the sole (which was not visible in input images) and the inside of the shoe (which was shadowed) are poorly reconstructed and are correctly identified as having a high uncertainty.

Figure 9: The 3D reconstruction of a rooster statue using image-based modelling (left column), two input images from approximately the same view point (middle column), and the mesh colour-coded with the uncertainty σ_{total} (right column). It can be seen that various sections of the underside of the model, which were shadowed and in some instances not visible on any of the input images, are poorly reconstructed and miss textures. Our algorithm correctly identifies these mesh sections as having a high uncertainty.

[3] D. Eberly. Distance between point and triangle in 3D. Available at `http://www.geometrictools.com/Documentation/DistancePoint3Triangle3.pdf/`. Last accessed on September 22nd, 2014.

[4] C. Früh and A. Zakhor. Constructing 3D city models by merging ground-based and airborne views. *IEEE International Conference on Computer Vision and Pattern Recognition*, 2:562–569, 2003.

[5] S. Izadi, D. Kim, O. Hilliges, D. Molyneaux, R. Newcombe, P. Kohli, J. Shotton, S. Hodges, D. Freeman, A. Davison, and A. Fitzgibbon. Kinectfusion: Real-time 3d reconstruction and interaction using a moving depth camera. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, UIST '11, pages 559–568, 2011.

[6] W. Matusik, C. Buehler, R. Raskar, S. Gortler, and L. McMillan. Image-based visual hulls. In *the* 27th *annual conference on Computer graphics and interactive techniques*, pages 369–374, 2000.

[7] M. H. Nguyen, B. Wünsche, P. Delmas, and C. Lutteroth. A hybrid image-based modelling algorithm. In *Proceedings of 36th Australasian Computer Science Conference (ACSC 2013)*, 2013.

[8] M. H. Nguyen, B. Wünsche, P. Delmas, C. Lutteroth, W. van der Mark, and E. Zhang. High-definition texture reconstruction for 3d image-based modeling. In *WSCG 2013 Full Paper, 24-27 June, Pilsen, Czech Republic, Union Agency*, 2013.

[9] S. Rusinkiewicz and M. Levoy. Efficient variants of the icp algorithm. In *Proceeding of Third International Conference on 3D Digital Imaging and Modeling (3DIM)*, pages 145–152, 2001.

[10] Q. Shan, B. Curless, Y. Furukawa, C. Hernandez, and S. M. Seit. Occluding contours for multi-view stereo. Available at `http://grail.cs.washington.edu/projects/sq_rome_g2/`. Last accessed on September 17th, 2014.

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