CPSC 601.13 - Course Project
Off-Axis Aligned Symmetric Reconstruction

Erika E. Harrison

April 30, 2012

Off-Axis Aligned Symmetric Reconstruction

Abstract

With the increase in computational power, advances within 3D reconstruction have improved significantly, thereby expanding to the previous unexplored fields of architecture, historical reconstruction and biological analysis. Merging the 3D point extraction of the Microsoft Kinect with the symmetric recognition of Mitra et al. [18], we can further advance computer vision algorithms in 3D reconstruction. By merging the concepts of these two, we can attempt to address occluded region patching through symmetric registration. The process presented herein simplifies the task to occluded region identification, symmetric patch recognition, patch deformation through displacement optimization, and finally symmetric patch application.

Introduction

Within the field of zoology, field researchers are required to reduce their impact on both the environment, and species for which they are exploring. To this end, motion-based cameras are being more prevalent within natural habitats [4]. From this information, computer vision techniques may be applied to build a full model through 3d reconstruction. Unfortunately, the information gained from the footage is often lacking, particularly when an opposing portion or patch of the specimen remains occluded from the camera. This limited information results in a reduced representation of an otherwise fully described specimen.

One approach to approximate the unseen portion of the specimen is to take advantage of the elements of bi-lateral symmetry which are common to most zoological species. In this way, a reasonable estimate may be stitched onto the occluded region using the visible portion. This is most readily observed through the off-axis aligned image taken of the frog in Figure 1. Notice how on the one side and whole of the face is visible. Through a reasonable guess - based on the symmetric nature of the head - the visible portion of the body could be used to estimate the occluded side.

Figure 1: Off-Axis Aligned Frog

In order to tackle this problem, we reduce it down to its simplist elements. With the advent of the Microsoft Kinect [14], we can assume that depth-based information, and therefore coarse 3d reconstructed estimations are plausible for data extraction. This replaces the otherwise two-dimensional input from current motion-based cameras. As such, we work with existing 3d meshes.

Furthermore, the Symmetrization algorithms proposed by Mitra et al. [17, 18] ensure that a resulting mesh can be adjusted to emphasize the symmetric elements. This includes a deformation to align a shape that exhibits bilateral symmetry. As such, we simplify our mesh to be relatively symmetrically aligned. Note that this encourages symmetry of shape, but not necessarily of geometry within the mesh. Consequently, we try to use mesh which are not perfectly symmetric.

From this simplification, we remove a region which could be denoted as occluded, or not observable from the capture device. In order to repair this hole, we must find a reasonable region - based upon the plane of symmetry, - deform the patch to fit the hole, and merge accordingly.This deformation, in accordance with the requirements of the course project, undergoes least squares optimizations in order to minimize deviation of the patch from the boundary to be filled.

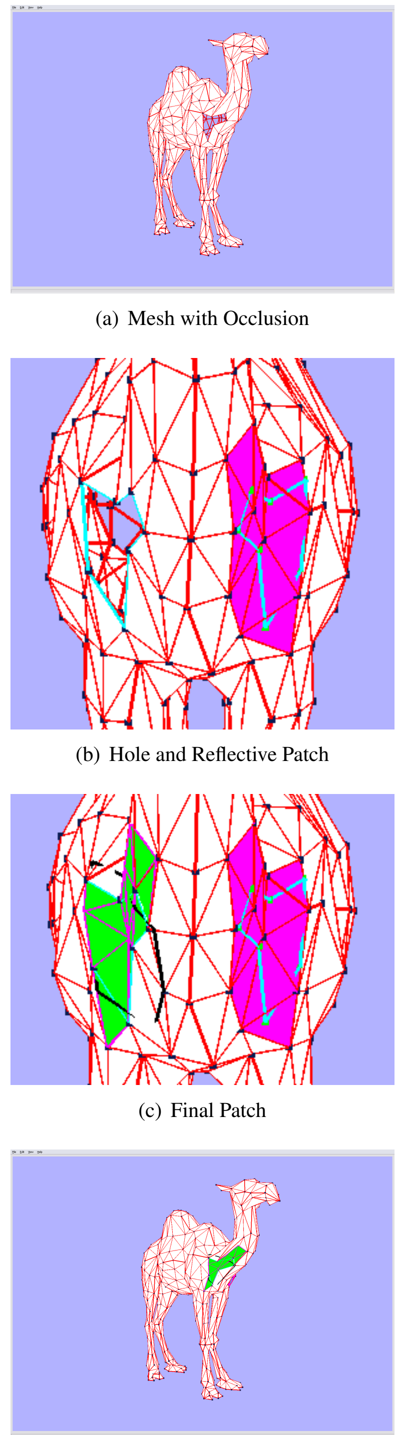
The details are described in the following sections. A preliminary discussion background and related work is explored. This is followed by a discussion on the methodology. Results, a discussion on future work, and conclusions finally wrap up the paper.

Background and Related Work

With the increasing processing power of graphics cards, and computers in general, the area of 3D reconstruction is becoming increasingly feasible. The possibilities have been further expanded with the recent accessibility of depth-based information through the relatively inexpensive Microsoft Kinect [14]. This technology provides depth information, supplementing the previously image-based explorations of Shape from Shading [2, 26, 28], Shape from Multiple Images [19,21] and Shape from Motion [5, 27]. This depth information has the potential to act as an imperative validating step, speeding up camera calibration [9] and improving the real-time quality of reconstruction.

The work within 3D reconstruction is being applied to a variety of important fields. Architectural [15] and historical researchers [10] have already begun to apply it towards non-invasive modeling and manipulation of existing forms. The application of such reconstruction towards biological and bio-medical analysis renders vast possibilities, such as analytical un-biased analysis of specimen and patients. Rudimentary work has only just begin, including the modeling of turtle shells [7], bird and bee flight [16, 22], as well as bat movement [12]. These explorations give researchers, and industry the potential to track species within crop fields [13], cages [3] and ensure optimal penning of cattle [6]. Pushing these works, and the research of 3D reconstruction has the potential to provide whisker-fine detail for physiological as well as behavioural analysis.

As alluded to, handling occluded regions within 3D reconstruction is still a field of exploration. A camera, depth-capturing or traditional, only sees one side of an object at any given time. For passive capture, such as cameras tracking species in wildlife reserves [8], one risks being unable to ever see the entire model. A reasonable solution is to assume a symmetric reconstruction which may be patched onto the missing region. While much work on occluded region patching has been explored within the 2D domain [11,24,25], the only work found to date on resolving 3D mesh occlusions is the symmetric work of Thrun and Wegbreit [23]. Their work relies on an exploration through the symmetric search space, and does not exploit the symmetric measure [18, 25] of the shape.



(a) Mesh with Occlusion

(b) Hole and Reflective Patch

(c) Final Patch

(d) Final Mesh

Figure 2: Steps for Symmetric Patching

Methodology

Basing the work on a relatively symmetric existing mesh, we are able to quickly explore the problem of patching through symmetric extraction. Furthermore, with a readily symmetric mesh, the symmetric region can be easily determined through obvious lines of bi-lateral symmetry. Since the underlying objective tends to work with specimens exhibiting bi-lateral symmetry, this seems a reasonable simplification.

From the mesh, we then remove an existing portion. This will represent our occluded region on an otherwise real-time constructed mesh. (See Figure 2(a)).

We then identify both the boundary of the hole, as well as the collections of faces that are more closely associated with the original boundary. Implementation is based on nearest point on the mesh, followed by the associated face and best visualized by the highlighted reflective patch in Figure 2(b). Light blue indicates the identified hole, which is reflected across the mesh. The bright purple faces highlight the faces influenced by the reflected boundary. These make up the faces which will become part of the resulting patch.

This collection of faces must then be deformed in order to fit readily within the original hole. A reasonable approach involves the minimization of deviation from the reflective boundary.

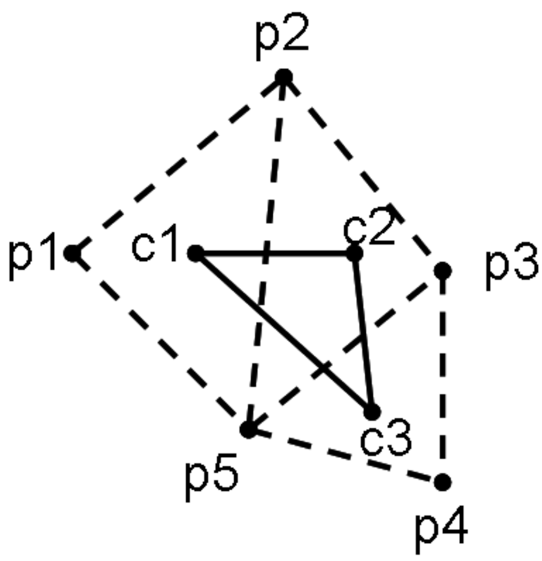


Figure 3: Example Mesh

For example, say we have a simple triangle boundary as illustrated in Figure 3, denoted by the vertices c_1, c_2, c_3. These have been reflected onto the faces - in dashed lines - and bounded by the vertices p_1 to p_5. In order to reduce devaition from the copying patch, we wish to select $c_i = p_i$ for all i. In particular, if we select it so that:

$$min \sum (c_i - c_i*),$$

then this gives us a reasonable collection of vertices - which can be re-arranged into a network of faces - upon which to fill the hole with. Once the p_i are associated with the c_i, we can reposition the vertices to coincide with c_i, and reduce the distortions on the patch when it is deployed.

We must keep in mind that in order to support our patch filling, we have additional constraints. For the system of equations - from our example in Figure 3, we observe that:

$$\begin{pmatrix} c_{1,1} & c_{1,2} & 0 & 0 & c_{1,5} \\ 0 & c_{2,2} & c_{2,3} & 0 & c_{2,5} \\ 0 & 0 & c_{3,3} & c_{3,4} & c_{3,5} \end{pmatrix} \begin{pmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5 \end{pmatrix} = \begin{pmatrix} c_1 \\ c_2 \\ c_3 \end{pmatrix}^T .$$

Here, we must have $c_{i,j} = \{0, 1\}$, boolean values 0 or 1. This corresponds to the binary nature of the vertex selection - either an existing vertex has been selected, or it hasn't. Furthermore, we require $\sum_j c_{i,j} = 1$, indicating that we must replace every c_i vertex with a p_i vertex. Lastly, we have that $\sum_i c_{i,j} = \{0, 1\}$, indicating that a p_i may be used no more than once to represent a c_i vertex. In this way, we avoid having a vertex expand into an edge, thereby constructing a supplementary face.

If we has stated our coefficients $c_{i,j}$ are rational numbers, than the system would simply solve to the barycentric coordinates for each of the points c_i. However, this would results in a large collection of faces that need to be cropped, split, and adjusted to accomodate this approach. Furthermore, one would need to explore the path between vertices to ensure that the resulting faces are constructed appropriately. We aim to simplify the process by snapping to existing vertices.

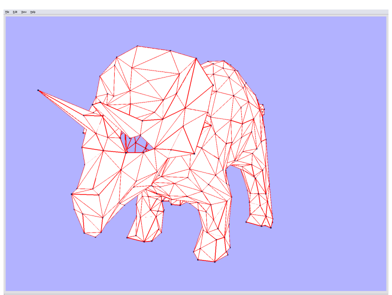
Unfortunately, discussions on handling boolean variants of least squares optimization problems are not well discussed. With this approach defined for solution finding, slight modifications have been employed to ensure a reliable answer. Verification against possible combinations of coefficients has been incorporated within the final solution.

With the reflective points p_i associated and repositioned to align with their corresponding c_i vertices, the selected faces are identifed as our replacement patch. These undergo the inverse reflective transformation to return to the original hole location, and adjusted to fit accordingly. This final step is illustrated in Figure 2(d). Bright green illustrates the resulting patch of deformed faces to fit the original hole. A black outline visualizes where the original boundary (p_i vertices) from the reflected side would have been transformed onto.

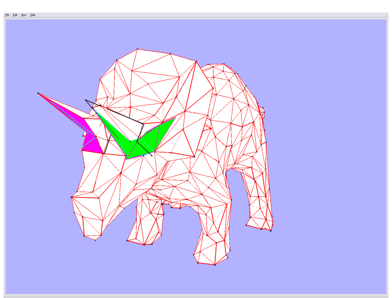
The resulting project has been constructed using Windows XP and Microsoft Visual Studio 9, under C++, OpenGL and Nokia's Qt interface. The Computational Geometry Library (CGAL) [1] as well as the Point Cloud Libraries [20] to hasten implementation for mesh management, manipulation and matrix computations.

Results

A couple examples are illustrated. From the occluded portion of the triceratops' horn (Figure 4), we observe a decent cover from the associated patch. Furthermore, occlusion from the back leg of a cow is illustrated. Recall that the black outlines the border for the reflected boundary before it is modified to fit the original hole. Furthermore, it is worth noting that some outlying faces may be currently carried over from the reflective patch. This has been left for future work.



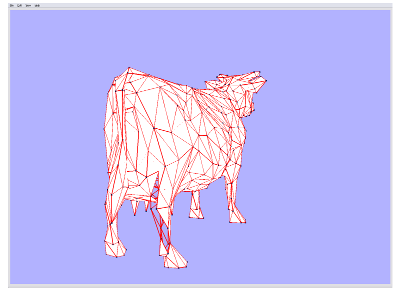
(a) Mesh with Occlusion



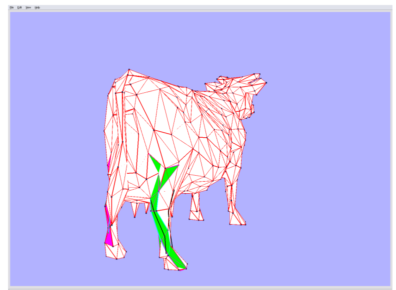
(b) Final Patch

Figure 4: Horn of Triceratops

There are a few limitations for working with the system. A triangle mesh with a defined symmetric plane must be employed. To change the symmetric axis, some parameters must be tweak and the program re-compiled. Faces on the interior of a patch are not well handled. This should be extended to grow the patched region to span all parts that should cover the hole. Finally, when the resulting boundary is created from the reflective patch, extraneous faces should be removed, and vertices which do no lie on the border be snapped to their incident edge. This will ensure a more compact patch.



(a) Mesh with Occlusion



(b) Final Patch

Figure 5: Cow Legs

Future Work

There are a number of natural and exciting extensions from the work presented. Drawing from the limitations of the prior section, patch growing, and resultant patch cropping are fairly straightforward. Support for arbitrary mesh styles - such as quadrilaterals, or irregular faces, is preferred, but not a conceptually interesting exploration. Numerous triangulation algorithms exist to rectify this situation.

More interesting extensions delve into the symmetric region identification. A formal implementation of the Symmetrization algorithm [18] would be of particular interest. From this, deformed yet symmetric shapes may be supported. This is particularly useful for the objective of supporting arbitrary specimen identification. An interesting theoretical extension is the use of non-reflective symmetry groups. Scaling, translation and rotational symmetries could as easily be supported with the Symmetrization implementation, thereby increasing the potential for symmetric occlusion patching.

Conclusion

Employing advancing technologies against existing problems offers the potential to further our understanding of the world around us in countless ways. By incorporating symmetric recognition for occlusion handling within 3D reconstruction, we can improve an initial estimate of the shape we are evaluating. This, in turn, offers us the potential to incrementally improve our approximation of the shape, or specimen, as more information is attained. A less intrusive approach to identification and analysis - offered by such 3D reconstruction techniques - aids biological researchers while reducing experimental interference.

With some adjustments to incorporate existing algorithms with the optimization technique employed here, we can more readily support a range of 3D occluded data. The rapidly expanding exploration of the Kinect's reconstructive potential, and potential occlusions, can only benefit from such efficient occlusion handling processes as those presented.

In the work presented, we have demonstrated a previously unexplored approach to mesh patching. Coupling directly off the shape of the visible portion, it could be improved through an improved symmetric representation, as explored by Mitra et al. [18]. Such improvements can act as a step towards a larger objective of capturing, representing and analyzing three dimensional data. Such a project would have been inconceivable only a few years ago, yet is within our reach through the speed and storage of modern computers. Unless we revisit these rudimentary or unexplored problems, these capacitative benefits will be for naught.

Bibliography

[1] "CGAL, Computational Geometry Algorithms Library," http://www.cgal.org.

[2] J. J. Atick, P. A. Griffin *et al.*, "Statistical Approach to Shape from Shading: Reconstruction of Three-Dimensional Face Surfaces from Single Two-Dimensional Images," *Neural Comput.*, vol. 8, pp. 1321–1340, August 1996.

[3] A. Atli, M. Gullu *et al.*, "Tracking the Rotation of Lab Animals via Computer Vision," in *Signal Processing and Communications Applications, 2007. SIU 2007. IEEE 15th*, June 2007, pp. 1–4.

[4] M. Bardo, "How Do You Track a Wolf Pack?" *BBC Nature*, April 2012.

[5] C. Bregler, A. Hertzmann *et al.*, "Recovering Non-Rigid 3D Shape from Image Streams," 2000, pp. 690–696.

[6] Z. Butler, P. Corke *et al.*, "From Robots to Animals: Virtual Fences for Controlling Cattle," *Int. J. Rob. Res.*, vol. 25, pp. 485–508, May 2006.

[7] Y. Chiari, B. Wang *et al.*, "Using Digital Images to Reconstruct Three-Dimensional Biological Forms: A New Tool For Morphological Studies," *Biological Journal of the Linnean Society*, vol. 95, no. 2, pp. 425–436, 2008.

[8] I. Cuthill, "Field Experiments in Animal Behaviour: Methods and Ethics," *Animal Behaviour*, vol. 42, no. 6, pp. 1007–1014, 1991.

[9] A. J. Davison, I. D. Reid *et al.*, "MonoSLAM: Real-Time Single Camera SLAM," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, pp. 1052–1067, June 2007.

[10] N. Futragoon, A. Kitamoto *et al.*, "3D Reconstruction of a Collapsed Historical Site from Sparse Set of Photographs and Photogrammetric Map," in *Computer Vision - ACCV 2010 Workshops*, ser. Lecture Notes in Computer Science, R. Koch and F. Huang, Eds. Springer Berlin / Heidelberg, 2011, vol. 6469, pp. 286–295.

[11] D. Hoiem, A. Stein *et al.*, "Recovering Occlusion Boundaries from a Single Image," in *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, Oct. 2007, pp. 1–8.

[12] G. Hu, H. Li *et al.*, "Fluorescent Optical Imaging of Small Animals Using Filtered Back-Projection 3D Surface Reconstruction Method," in *BioMedical Engineering and Informatics (BMEI). International Conference on*, vol. 2, May 2008, pp. 76–80.

[13] M. Israel, "A UAV-Based Roe Deer Fawn Detection System," H. Eisenbeiss, M. Kunz *et al.*, Eds., September 2011.

[14] S. Izadi, D. Kim *et al.*, "KinectFusion: Real-Time 3D Reconstruction and Interaction Using a Moving Depth Camera," in *UIST'11*, 2011, pp. 559–568.

[15] N. Jiang, P. Tan *et al.*, "Symmetric Architecture Modeling with a Single Image," *ACM Trans. Graph.*, vol. 28, pp. 1–8, December 2009.

[16] M. Maurer, M. Rther *et al.*, "Rigid body reconstruction for motion analysis of giant honey bees using stereo vision," 2010.

[17] N. J. Mitra, L. J. Guibas *et al.*, "Partial and Approximate Symmetry Detection for 3D Geometry," *ACM Trans. Graph.*, vol. 25, pp. 560–568, July 2006.

[18] ——, "Symmetrization," *ACM Trans. Graph.*, vol. 26, pp. 1–8, July 2007.

[19] A. Mulayim, U. Yilmaz *et al.*, "Silhouette-Based 3-D Model Reconstruction From Multiple Images," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 33, no. 4, pp. 582–591, Aug. 2003.

[20] R. B. Rusu and S. Cousins, "3D is here: Point Cloud Library (PCL)," in *IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, May 9-13 2011.

[21] S. Seitz, B. Curless *et al.*, "A Comparison and Evaluation of Multi-View Stereo Reconstruction Algorithms," in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, vol. 1, June 2006, pp. 519–528.

[22] A. D. Straw, K. Branson *et al.*, "Multi-Camera Real-Time Three-Dimensional Tracking of Multiple Flying Animals," *Journal of The Royal Society Interface*, 2010.

[23] S. Thrun and B. Wegbreit, "Shape from Symmetry," in *Proceedings of the Tenth IEEE International Conference on Computer Vision - Volume 2*, ser. ICCV '05. Washington, DC, USA: IEEE Computer Society, 2005, pp. 1824–1831.

[24] M. Venkatesh and S.-E. Cheung, "Symmetric Shape Completion Under Severe Occlusions," in *Image Processing, 2006 IEEE International Conference on*, Oct. 2006, pp. 709–712.

[25] H. Zabrodsky, S. Peleg *et al.*, "Completion of Occluded Shapes Using Symmetry," in *Computer Vision and Pattern Recognition, 1993. Proceedings CVPR '93., 1993 IEEE Computer Society Conference on*, June 1993, pp. 678–679.

[26] G. Zeng, Y. Matsushita *et al.*, "Interactive Shape from Shading," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1, June 2005, pp. 343–350.

[27] L. Zhang, B. Curless *et al.*, "Spacetime Stereo: Shape Recovery for Dynamic Scenes," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, June 2003, pp. 367–374.

[28] R. Zhang, P.-S. Tsai *et al.*, "Shape-From-Shading: A Survey," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 21, no. 8, pp. 690–706, Aug 1999.