Computer Vision 2
Final Assignment: 3D Mesh Generation and Texturing.

Riaan Zoetmulder
MSc Artificial Intelligence
University of Amsterdam
riaanzoetmulder@gmail.com

Minh Ngo
MSc Artificial Intelligence
University of Amsterdam
nlminhtl@gmail.com

1 Depth-based and Texture-based 3D Reconstruction Comparison

We firstly discuss the advantages and disadvantages of depth based 3D reconstruction methods. The depth based ICP approach finds pairs of corresponding points and attempts the rotation and translation between them. One of the benefits is that it always converges to a local optimum. Furthermore, if efficient subsampling is used, such as normal shoot, it can converge quickly. A disadvantage of ICP is that it must have a good initial starting point, otherwise it will converge into a bad local optimum. Furthermore, the basic version of ICP is sensitive to outliers, because it incorporates all the points in the objective function that it minimizes (Phillips et al., 2007).

Secondly, we discuss the advantages and disadvantages of Structure from Motion (SfM). An advantage of the SfM algorithm is that it can handle a large number of views and it can handle missing data by subsampling dense blocks. A disadvantage is that parameters will grow as the amount of views grows, thus it will take a longer time as the data becomes larger and it requires a good initial condition. In addition it relies on existence of interest points that depend on the surface texture information.

We can combine ICP with SfM by using the ICP algorithm to find the best transformation from the 3D point cloud generated from one set of images to 3D point cloud generated from another set of images instead of using Procrustes Analysis (Zinßer et al.).

2 3D Meshing and Watertighting

To merge point clouds from multiple frames we first threshold an object by depth to remove a background. We estimate normals and transform them to 1 coordinate system knowing the camera pose. After having a single point cloud with estimated normals we subsample it using Pass Through filter and run the Poisson Triangulation algorithm to obtain a polygon mesh. We don't consider Marching Cubes Approach because of the high memory consumption and far-to-real-time performance. For completeness we briefly discuss the Marching cubes approach and the Poisson algorithm below.

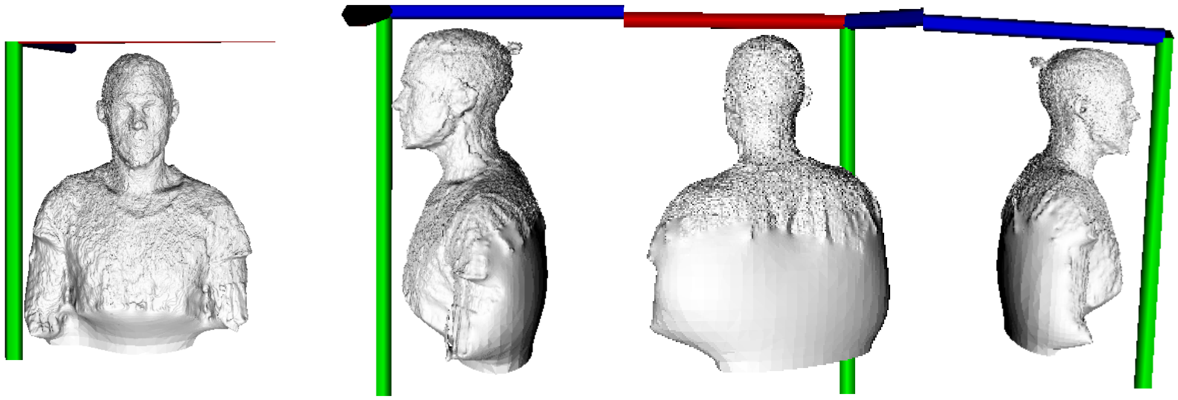

Figure 1: Reconstructed meshed model.

Poisson Triangulation

The Poisson triangulation algorithm works in the following way:

Firstly, you set the octree. An octree is a search tree for searching through three dimensional space, where each leaf represent a cube. Secondly, the vector space is calculated. Thirdly, the indicator function is computed followed by the computation of the poisson function. Fourthly, the iso-surface is extracted.

The resolution of the solution obtained from this algorithm depends on the depth of the octree. The deeper the octree, the higher the resolution. Furthermore, the algorithms memory and time requirements are quadratic. A downside to this algorithm is that when there are no samples of a certain area, it may wrongfully correct certain regions (Kazhdan et al., 2006).

Marching Cubes

The marching cubes algorithm tries to draw planes between interpolated values along the boundaries of a cube. 256 different representations can be specified for the cube. Some cases are ambiguous and several additional cases are added to the possibilities for this.

The algorithm divides the point cloud into cubes. Depending on how small the cubes are a coarse or fine resolution can be obtained. When the algorithm finishes, the union of the planes is the mesh. Usually this mesh is watertight and the time complexity is cubic (Bartz and Meissner, 2000).

3 Texturing 3D Model

To texture the 3D model we extract a point cloud corresponding to the generated polygon mesh. We are performing a texturing algorithm iteratively over frames. For each frame we transform the point cloud back using a inversed frame's camera pose matrix.

We have implemented a zbuffer approach to determine visible triangles. In our approach we assume a view window to be a size of the depth map. For each of (x, y) coordinates of a view window we determine a closest point in the Z axis. Since we have sampled points to do meshing there is a chance to have only points from invisible (back surfaces) for the certain (x, y) coordinate, therefore we have introduced a second stage. We are passing through the zbuffer and check the difference between Z coordinates of neighbor points. If the difference is higher than a threshold then only the closest Z point is taken in consideration.

For visible triangles we compute u, v coordinates and match them with colors available from RGB images. If u, v are outside a range between 0 and 1 then we ignore a point since it's outside a view frame. If we find several matched colors from several frames then we average RGB values, if we don't find any matches then a color of a nearest colored triangle is taking into account.

4 Results

Results obtained after the texturing stage can be seen in the figure 2. Texture has been recovered precisely. However there is an issue with textures set for uncovered regions caused by an introduced algorithm that takes closest euclidean colored point into account (instead of the distance on a surface itself). The further improvement requires a fast data structure to do Depth-First Search among neighborhood triangles.

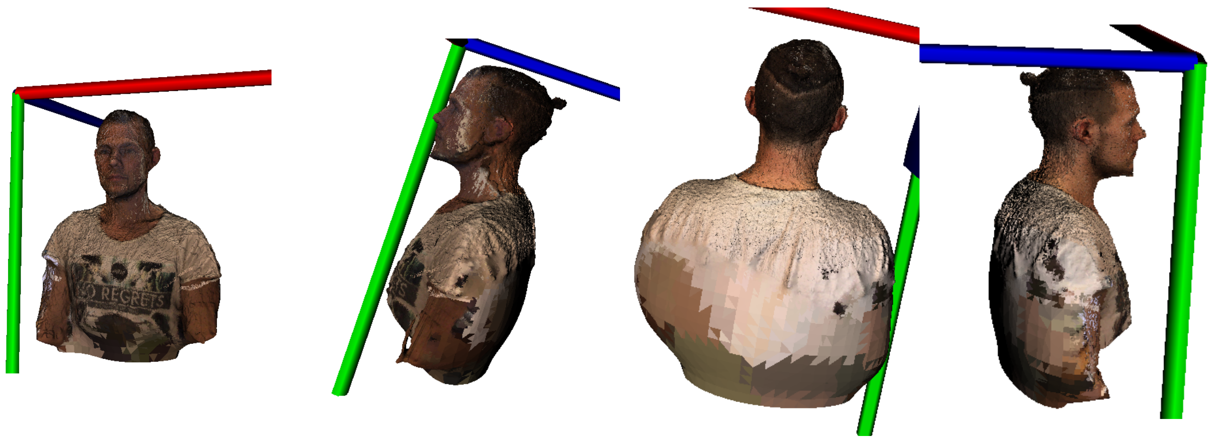

Figure 2: Reconstructed and textured model. ”No regrets” can be easily detected.

References

D. Bartz and M. Meissner. Voxels versus polygons: A comparative approach for volume graphics. 2000.

M. Kazhdan, M. Bolitho, and H. Hoppe. Poisson surface reconstruction. In *Proceedings of the Fourth Eurographics Symposium on Geometry Processing*, SGP '06, pages 61–70, Aire-la-Ville, Switzerland, Switzerland, 2006. Eurographics Association. ISBN 3-905673-36-3. URL http://dl.acm.org/citation.cfm?id=1281957.1281965.

J. M. Phillips, R. Liu, and C. Tomasi. Outlier robust icp for minimizing fractional rmsd. In *3-D Digital Imaging and Modeling, 2007. 3DIM'07. Sixth International Conference on*, pages 427–434. IEEE, 2007.

T. Zinßer, J. Schmidt, and H. Niemann. Performance analysis of nearest neighbor algorithms for icp registration of 3-d point sets.