Craniofacial variation III: Efficient, landmark-free superimposition of head surface scans

Benjamin Pomidor[1], Brian Corner[2], Dennis Slice[1]

[1]Department of Scientific Computing
Florida State University
[2]US Army Natick Soldier Research
Development and Engineering Center

Abstract The now-familiar approach of landmark-based morphometric analysis is problematic for many anthropological (and other) data sets. Many structures of interest lack meaningful and repeatably identifiable landmarks. At the same time, it is becoming increasingly easy to collect data via surface scans for ever larger samples of specimens, making manual landmark placement for superimposition and other analyses impractically time consuming. One way to address this is through the use of landmark-free methods, such as the Iterative Closest Point (ICP) algorithm, to directly process the surface scans. ICP relies on a nearest neighbor search (NNS), which is very computationally intensive. A simple NNS requires on the order of n^2 operations, where n can reach easily into the hundreds of thousands. This poster presents a computationally efficient approach to ICP that yields a substantial reduction in computation costs. In this efficient method, we cluster the vertices in one scan into small groups, tracking cluster creation to form a spatial tree. Searches performed with these trees dramatically decrease the computational cost, to order $n * \log(n)$. For instance, superimposing two scans of 23,000 points using the new process was approximately sixty times faster than the naive method. The algorithm has been demonstrated here by fitting a sample of 40 bald head scans, provided by the US Armed Services, to a reference template to show regional variability. While the ICP algorithm is dependent on initialization, the conventional coordinate axes in the scans provide adequate initialization parameters. This method is applicable to most landmark-sparse structures.

This work was funded, in part, by Cooperative Research Agreement W911QY-12-2-0004, between Florida State University and the US Army Natick Soldier Research, Development, and Engineering Center. Approved for public release.

Introduction

- Traditional superimposition requires time-consuming manual landmark placement.
- Many objects of interest do not have readily apparent landmarks on smooth regions
- 3D scanners are powerful data collection tools that generate surface scans comprised of hundreds of thousands of individual points, resulting in very large amounts of data.
- The Iterative Closest Point algorithm (ICP) can superimpose surface scans without landmarks.

Data Processing

- Before applying any kind of algorithm, we must consider the scans produced by the 3D scanner.
- Most raw scans have hundreds of thousands of points. Processing all of these points is time consuming for any algorithm.
- To reduce computational time, we can decimate scans. Because of the density of the points in the original scan, doing so will not significantly reduce detail.

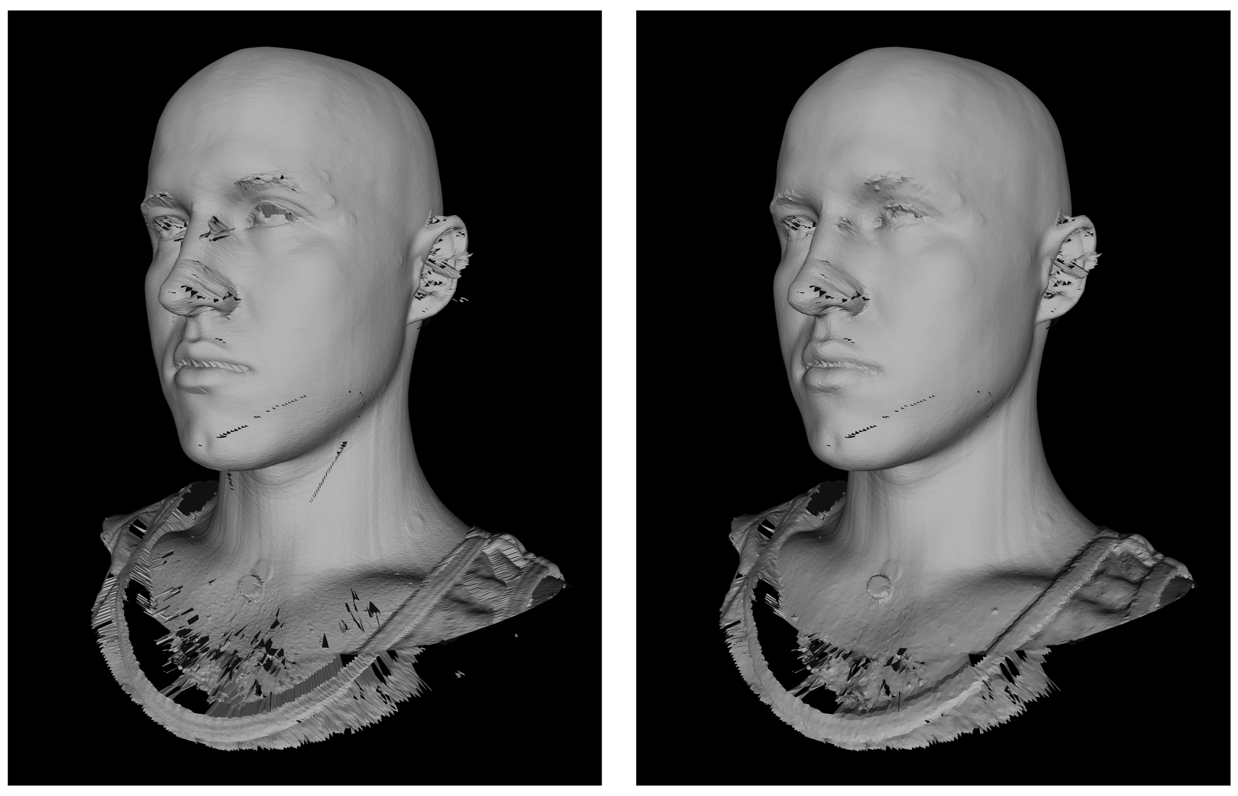

Figure 1: Left: Before decimation (456988 points). Right: After decimation (115836 points).

- Casual inspection of Figure 1 confirms this retention of detail, despite the decimated scan only having about one quarter the number of points.

Iterative Closest Point Algorithm

- The steps in the ICP algorithm:

1. Pair each point in a scan with its nearest neighbor in the other scan.
2. Use point pairs to calculate the transformation to minimize the distance between surfaces.
3. Repeat until convergence is reached.

- The ICP algorithm is very dependent on initialization, but the conventional axes from the scanner are typically sufficient given consistent scan protocol.

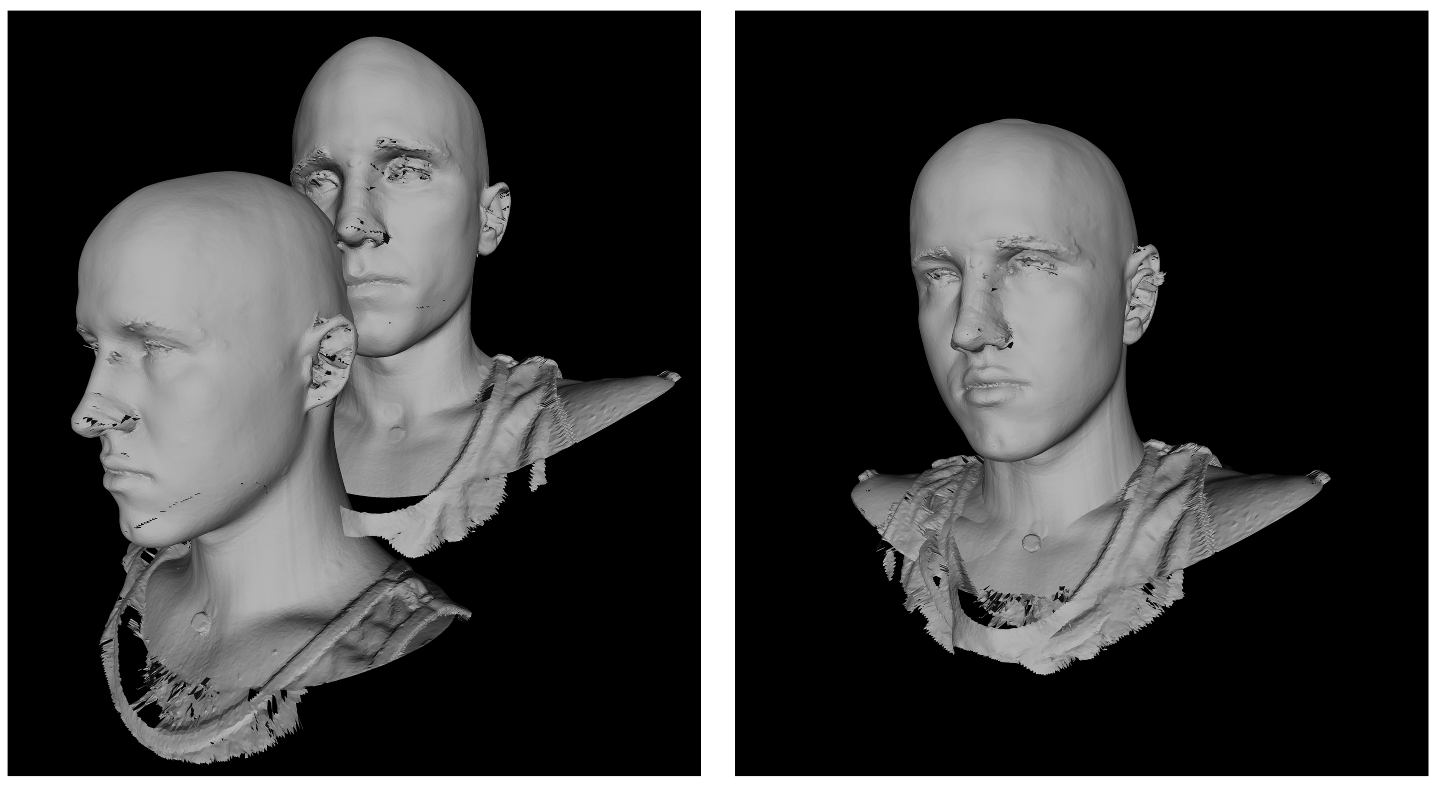

Figure 2: Left: Two head scans before ICP is applied. Right: After ICP superimposition.

- Figure 2 shows the superimposition achieved by the ICP algorithm. Note especially the ears and shoulders for evidence of superimposition.

Computational Time

- The main drawback of the ICP algorithm is the nearest neighbor search, which, naively, requires order n^2 operations.
- We can address this computational bottleneck by using a Binary Space Partitioning method, specifically a Bisecting K-Means tree.
- This is a modified Bisecting K-Means clustering algorithm that tracks the creation of subnodes to make a spatial tree, which we can use to find nearest neighbors much more efficiently.
- Using this tree reduces computational time from order n^2 to order $n * \log(n)$. In practice this means reducing processing time from more than seventy-five minutes in the naive method to less than four seconds in the optimized method for scans of approximately 110,000 points.

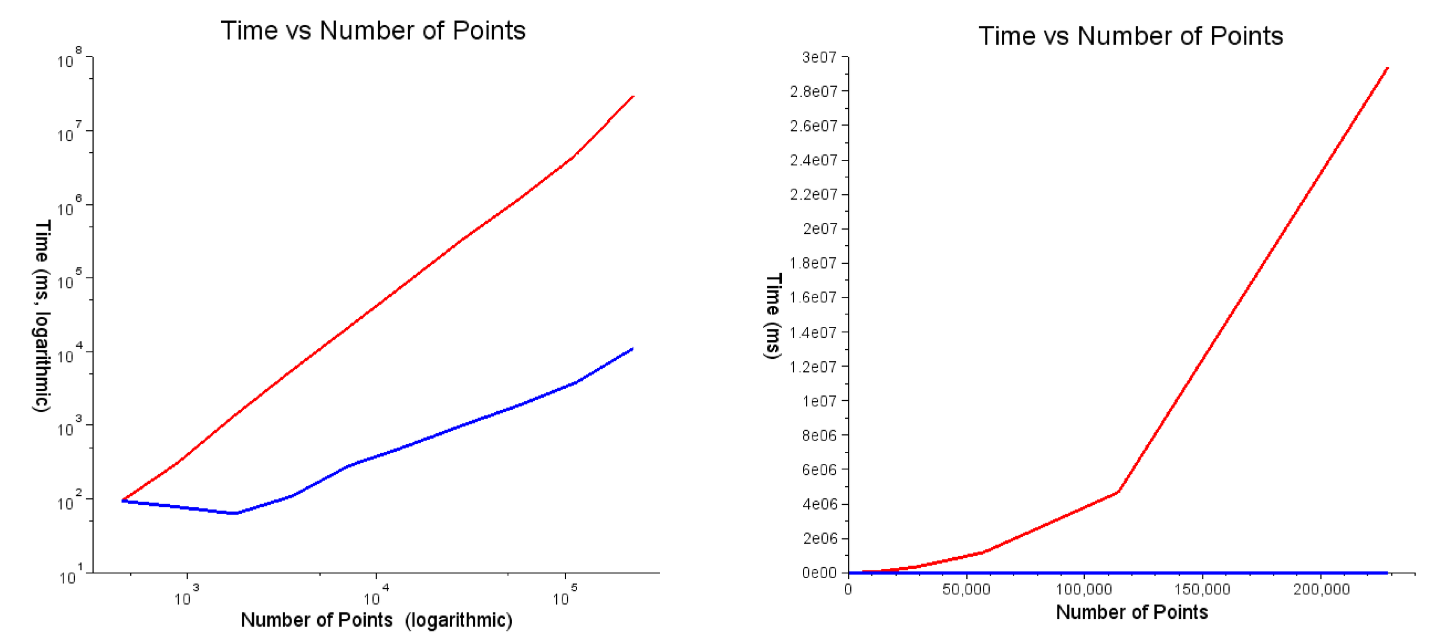

Figure 3: Graphs showing the difference in computational time for n points using the naive method (red) and the optimized method (blue). Tests performed on an Intel Core i5 2.50 GHz processor.

Superimposition

- By reducing the computational time, it is possible to superimpose large sample sizes in a short amount of time on a regular desktop computer.
- Once superimposed, we can use the pairwise matches to calculate the average head shape of the sample. We are also able to calculate the variance in position at each point, which we visualize in Craniofacial Variation IV.

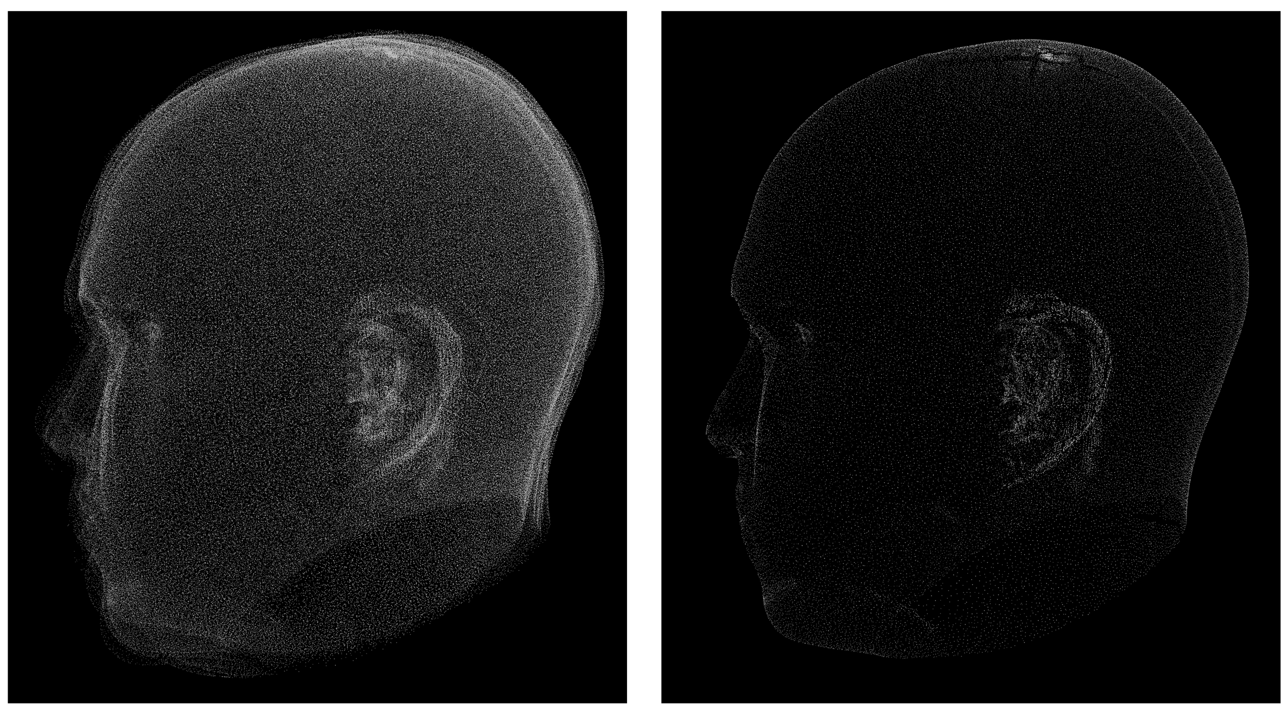

Figure 4: Left: The points from several head scans superimposed. Right: The calculated average head.

Results and Conclusion

- In testing, the optimized Java implementation was able to superimpose 40 head scans and calculate their average in approximately four minutes on an Intel Core i5 2.50 GHz processor. For comparison, this same task with the naive method would take an estimated four days of computational time.
- The freedom from landmarks, relatively high speed, and pairwise matching, together, make the ICP algorithm a powerful alternative to traditional landmark-based superimposition.

References

- Chen, Y., Medioni, G., "Object Modeling by Registration of Multiple Range Images," *Proceedings of the 1991 IEEE International Conference on Robotics and Automation*, p. 2724-2729, 1991.
- Pettinger, G., Di Fatta, G., "Space Partitioning for Scalable K-Means," *Ninth International Conference on Machine Learning and Applications*, p. 319-324, 2010.