# Assignment 2 - Registration and reconstruction

Geometry Processing - Carleton University

Last update: September 23, 2024

This assignment has two parts, related to registration and reconstruction.

## 1 Part 1: Surface registration

Implement a program for point cloud alignment based on the RANSAC method. The method takes as input two points clouds $\mathcal{C}_1$ and $\mathcal{C}_2$ and returns a rigid transformation $T$ that aligns the point clouds as best as possible. The method can be described by the following pseudo-code:

**Algorithm 1** RANSAC point cloud alignment

**procedure** ALIGN($\mathcal{C}_1$, $\mathcal{C}_2$) ▷ Source and target point clouds $\mathcal{C}_1$ and $\mathcal{C}_2$
    **while** Number of iterations not finished **do**
        **Random sample:** select a set of candidate matches $M$;
        Derive a transformation $T$ from $M$;
        Apply $T$ to all points in $S$;
        **Consensus:** Find inliers $I$ to the alignment;
        **if** $|I|$ is better than the number of inliers for the current best **then**
            $T$ = Recompute the transformation from all inliers $I$
            current best = $T$
    Return current best

The set of candidates $M$ should contain three matches. Each match is composed of a point $P_1$ randomly sampled from $\mathcal{C}_1$ and a corresponding point $P_2$ from $\mathcal{C}_2$. To find the corresponding point, randomly sample a point from the top 20 points in $\mathcal{C}_2$ matched to $P_1$ according to the spin image descriptor. Experiment with different thresholds to determine the number of inliers and different numbers of iterations to find a configuration that leads to satisfactory results. You are free to decide any other implementation details of the method not discussed here. However, make sure that your choices are reasonable and that you follow the core ideas of the method (the implementation should include random sampling and consensus – RANSAC).

You are free to use any code/functions provided in the GeomProc library. If you reuse code from other sources, make sure to give credit and that the code does not implement the main task of the assignment, i.e., the RANSAC method itself.

When testing the method, start with a basic sanity check by aligning a point cloud to itself, without any transformations. Then, make the problem increasingly more difficult and evaluate the results: 1. Align a point cloud to a transformed version of itself, 2. Use two different point clouds sampled from the same mesh and differing by a transformation, and, 3. Add a small amount of noise to the points. To obtain a point cloud, you can sample points from an existing mesh using the sampling function in the GeomProc library. You can focus your experiments on one selected shape.

In your assignment report, provide the thresholds/parameters that you used in the experiments, and describe any aspects of the implementation where you had to make your own choices. Then, write a short analysis section where you compare the different test settings described above and show figures with visual results of the method, providing some conclusions on the performance of the method.

## 2 Part 2: surface reconstruction

Implement the interpolation-based surface reconstruction method of Kolluri [2005]. Given a set of points $\mathbf{p}_i \in \mathcal{C}$, the method computes the implicit function value for a 3D query point $\mathbf{x}$ based on the equation:

$$\Phi(\mathbf{x}) = \frac{\sum_i \phi(\|\mathbf{x} - \mathbf{p}_i\|)\, \mathbf{n}_i^T (\mathbf{x} - \mathbf{p}_i)}{\sum_i \phi(\|\mathbf{x} - \mathbf{p}_i\|)}, \tag{1}$$

where

$$\phi(r) = \exp(-r^2/\epsilon^2), \tag{2}$$

and $\mathbf{n}_i$ is the normal vector of point $\mathbf{p}_i$.

Provide the implicit function $\Phi$ defined in this manner to the marching cubes method in the GeomProc library to reconstruct triangle meshes. Specifically, try this method on a simple shape (e.g., a sphere) and a more complex shape (e.g., the bunny), and compare to the RBF method provided as a demo in the library. Experiment with different values for $\epsilon$. In your report, show the results that you obtained and discuss the visual differences to the RBF method.

Again, you are free to use any code/functions provided in the GeomProc library. It is probably a good idea to start your implementation from one of the reconstruction demos in the library.

## 3 Submission

Submit a zip file with your code and a single report discussing the results for both parts. Submit the report in pdf format. You don't need to include the

GeomProc library in your submission if you did not modify it. Include the input meshes that you used in your experiments if they are not the ones from the GeomProc library.

## 4 References

R. Kolluri, "Provably Good Moving Least Squares", Proc. of the Symposium on Geometry processing, 2004.