**Document for NICP (non-rigid iterative Closest Points) in our project**

This is the document for NICP in our project. This will show how NICP works and what changes I made for Our project to make this algorithm works better.

**Relative works:**

ICP (iterative closest points) is a common resolution to combine two mesh/points cloud from several scans of one object together.

There are three steps for Iterative closest points.

1. Find the registration

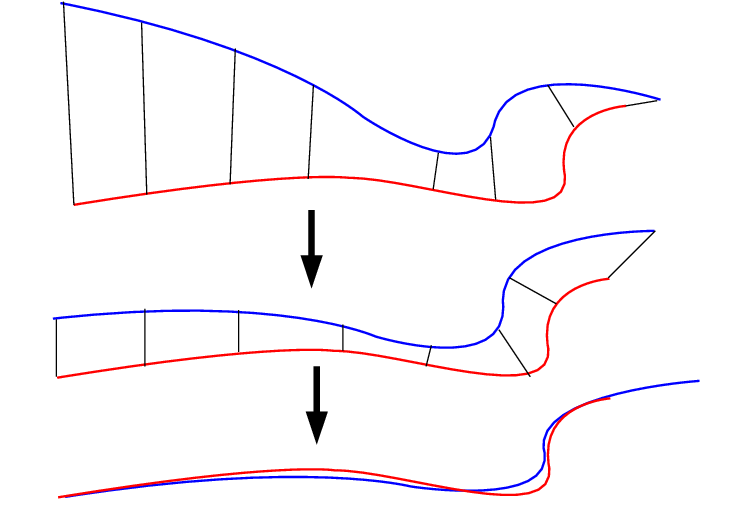
2. Find a transformation matrix to minimize the Cost function

3. Transform the original model in a rate.

1. Find the registrations

Registrations are the relation between the original, like the following pictures shows.

By default, each point on the source model will register the closest points on the target model. But if we have knowledge about the correspondence between two models, we can build up the registrations by ourselves.

[1]

2. Find a transformation matrix to minimize the Cost function

We have two sets of points {p} and {q}

If we use Least-squares fitting. We can use this formula to calculate the cost:

[2]

Dis = (|q – (Rp + T) |) ^2

R is the (3 \* 3) rotation matrix and T is the (3\*1) translation vector.

T can be calculated by difference between the centroid of two model [3].

While for R, we can calculate that by do singular value decomposition (SVD).

Let p’ = p – pc, q’ = q – qc, which q arrange in order given by step1. Pc and qc refer the centroid of the model. (pc = avg(p))

1. Let H = p’ \* q’ t (transpose of q’)

2. Find the SVD of H which H = UAVt

3. Calculate X = VUt

4. Calculate Det(X)

If Det(X) = +1 Then R = X

If Det(X) = -1, fails (usually not occur)

3.Apply transformation:

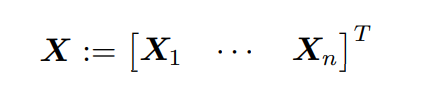
p = Xp

**Non-rigid Iterative Closest Points**

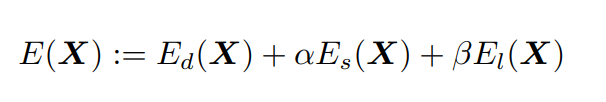
The difference between ICP and NICP is that in ICP all vertices share same transformation matrix while in NICP each vertex has its own transformation matrix.

The Steps are close to the NICP, First compute the registration then Compute transformations to minimize the cost function, which in NICP, we have N transformation matrices.[4]

Xi is the transformation matrix for pi 🡪 pi’, we can organize the matrices like following.

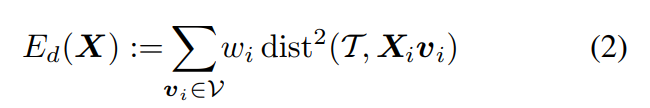


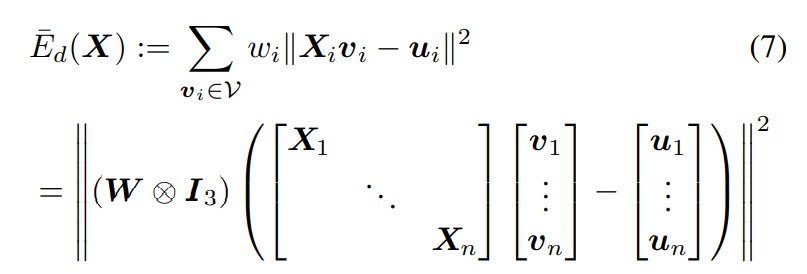
For cost function, there are three part of the cost function

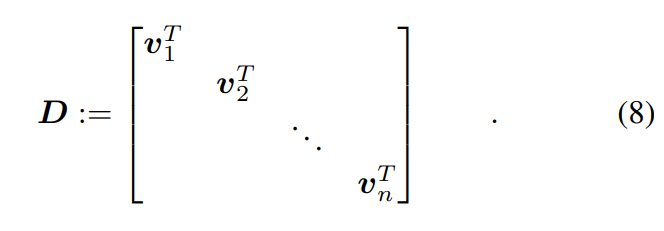


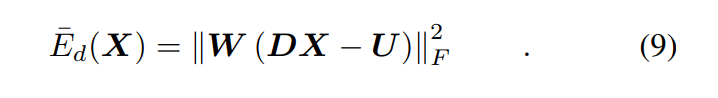
Ed is the distance; Es is the deformation penalty; El is landmark cost which we do not use in our project.

For Ed, is sum(norm(qi – Xi\*pi)), which means the distance between the original model and the deformed model.

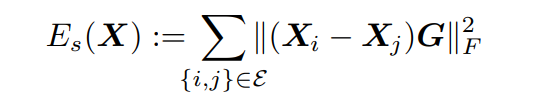


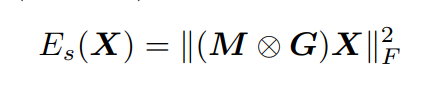






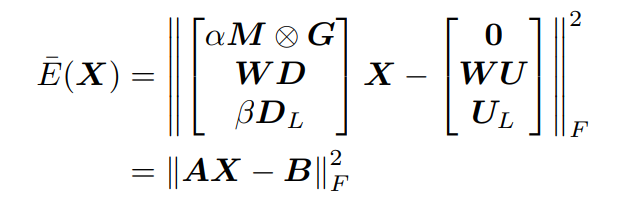
Es regularize the deformation. For example, there is a edge connect i and j. We want Xi and Xj keep the same so that to reduce the deformation. G is diag (1, 1, 1, gamma). Gammas represent the weight for rotational deformation and translational deformation.





(M is node-arc incidence matrix)

To minimize the cost, we can minimize norm of vector (sqrt (Ed), sqrt (Es), sqrt (El)). To satisfy our requirement, we can add more to this vector, as much as the matrix keep positive definite.



Become an ICP problem.

**Changes in our project**

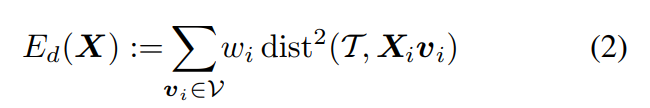
1. For registration step, rather than using the iterative registration, we use a registration from the reference fit.

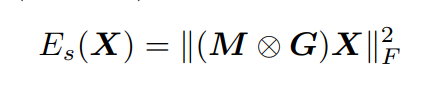
There are two advantages of this.

First, it compatible with morphable model because we want vertex with same index have same meaning. Meanwhile, it is more robust for some bad case which registration can be bad.

Second, for ICP and NICP, fit a more complete model with an incomplete target model may result in unexpected deformation, like fit a head to a face. By reversing the registration, can avoid this situation.

2. Smart Edges and vertices weight.





We set high weight for vertices on contours and edges on center of the face and verse visa. [5]

**Reference**

[1] Smistad, Erik & Falch, Thomas & Bozorgi, Mohammadmehdi & Elster, Anne & Lindseth, Frank. (2015). Medical image segmentation on GPUs - A comprehensive review. Medical Image Analysis. 20. 1-18. 10.1016/j.media.2014.10.012.

[2] K. S. Arun, T. S. Huang and S. D. Blostein, "Least-Squares Fitting of Two 3-D Point Sets," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-9, no. 5, pp. 698-700, Sept. 1987, doi: 10.1109/TPAMI.1987.4767965.

[3] T. S. Huang, S. D. Blostein and E. A. Margerum, "Least-squares estimation of motion parameters from 3-D point correspondences", Proc. IEEE Conf. Computer Vision and Pattern Recognition, 1986-June-24-26.

[4] B. Amberg, S. Romdhani and T. Vetter, "Optimal Step Nonrigid ICP Algorithms for Surface Registration," 2007 IEEE Conference on Computer Vision and Pattern Recognition, 2007, pp. 1-8, doi: 10.1109/CVPR.2007.383165.

[5] Ploumpis, Stylianos & Wang, Haoyang & Pears, Nick & Smith, William & Zafeiriou, Stefanos. (2019). Combining 3D Morphable Models: A Large scale Face-and-Head Model.