Estimation of Point Cloud Object Pose using Particle Swarm Optimization

Introduction

- Iterative closest point (ICP) algorithm is a popular algorithm due to its simplicity. It is demonstrated that the ICP algorithm can be used in practice well by optimizing the use of lean and simple descriptors.
- Iterative closest face (ICF) is able to converge to the solution without any initial knowledge on the pose of the target compared to traditional algorithms. However, it left some work to do to be applied into practice.

Iterative Closest Face Algorithm

- Judging the projection of the point is inside the CAD model face. Assuming there is a triangle of vertexes ABC and a point N:

$$[(B-A)\times(N-A)]\times[(N-A)\times(N-A)]\geq 0$$
$$[(A-B)\times(N-B)]\times[(C-B)\times(N-B)]\geq 0$$
$$[(A-C)\times(N-C)]\times[(B-C)\times(N-C)]\geq 0$$

- Algorithm objective function

$$F(P,S)=\sum_{m\in S} f(m,P)$$

$$f(m,P)=\left|\begin{matrix} 1-\frac{g_p(m)}{thr}, if g_p(m)<thr \\ 0 \quad if g_p(m)\geq thr \end{matrix}\right.$$

$$g_p(m)=\left|\begin{matrix} d_p(m) \quad if\ d_p(m)<dist \\ k\cdot d_p(m)\ if\ d_p(m)\geq dist \end{matrix}\right.$$

where $d_p(m)$ is the distance between point m and the closest face of the model aligned to pose P.

Particle Swarm Optimization

- The positions x_i and the velocities v_i of of particles are updating according to:

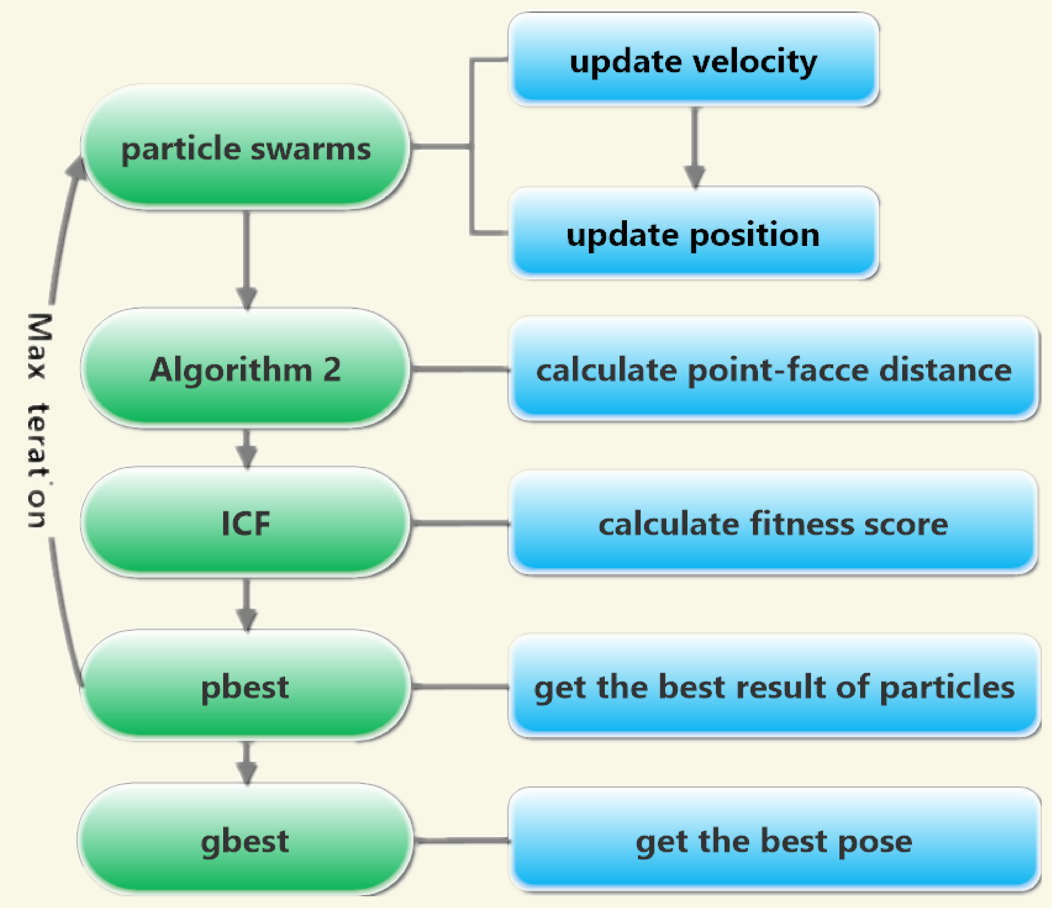
$$x_i(k)=x_i(k-1)+v_i(k)$$

$$v_i(k)=wv_i(k-1)+c_1\phi_1(pbest_i-x_i)$$
$$+c_2\phi_2(gbest_i-x_i)$$

- $c1$ is the acceleration constant for the cognitive component. $c2$ is the acceleration constant for the social component. The value $pbest$ and $pbest$ represent the best position of the particle and the best global position respectively.

Proposed Algorithm

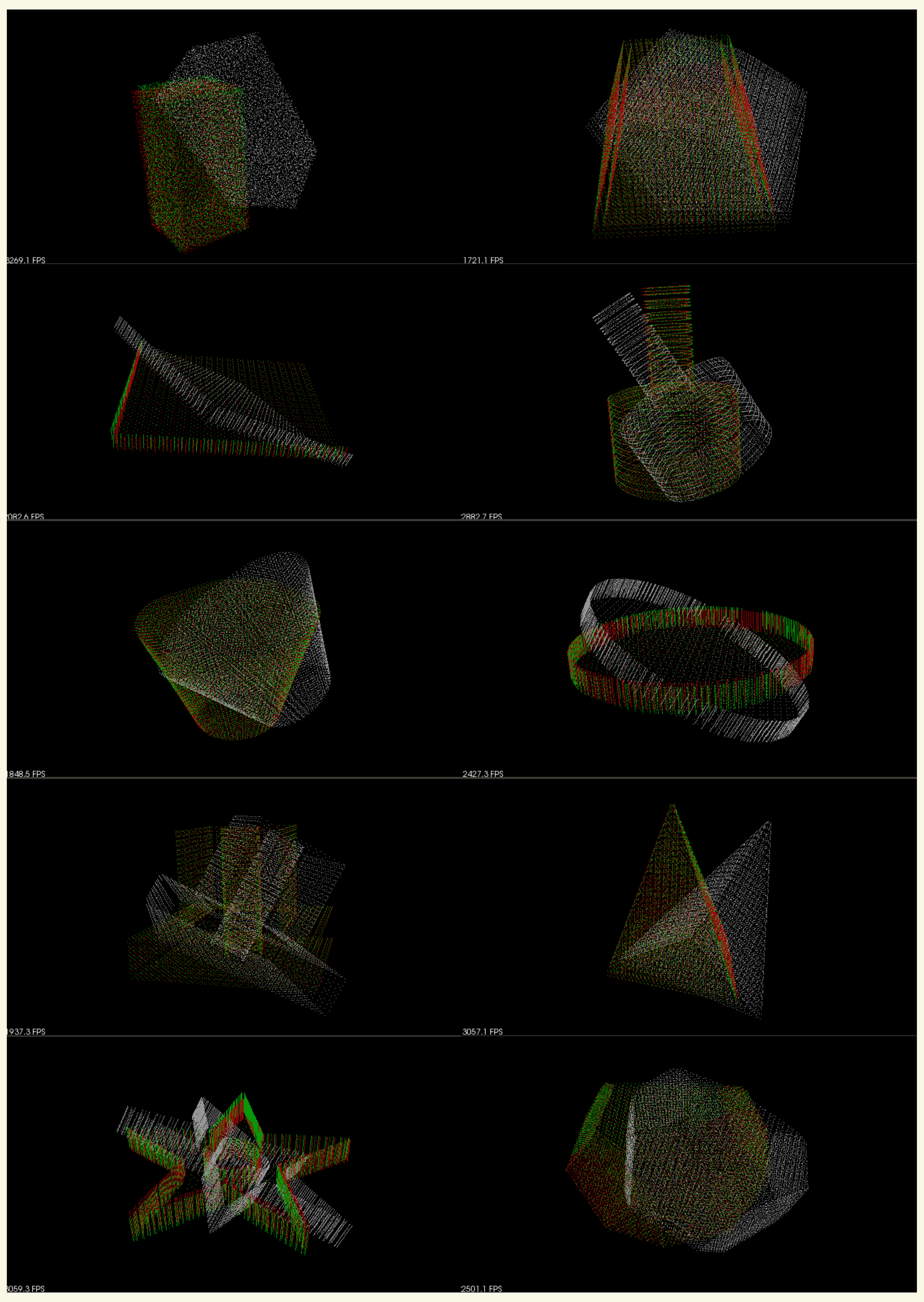
Based on the above method, we use particle swarm optimization to help search the best pose for ICF algorithm and the whole process is described as the flowchart:



Experiment Result

- The CAD models used are shown as follows:

- Visualization results of a set of experiments:The more accurate the searched results are, the more similar the standard pose (green) and the searched pose (red) will be.



- The experiment results of proposed algorithm:

	Transition/mm	Rotation/rad	E(T)/mm	E(R)/rad
cuboid	[0.2416, 0.0984, 0.1842]	[0.5413, 0.5971, 0.0512]	0.3322	0.0914(5.24°)
forprise	[0.3013, 0.2035, -0.1103]	[0.5317, 0.5097, 0.0060]	0.3051	0.0172(0.98°)
slab	[0.4653, 0.1683, 0.0021]	[0.5269 0.5245 0.0182]	0.1719	0.0185(1.06°)
hammer	[0.2114, 0.0624, -0.0473]	[0.5130, 0.5270, 0.1124]	0.2990	0.1130(6.47°)
circulumn	[0.0801, 0.0615, 0.0339]	[0.5314, 0.5155,0.0103]	0.4257	0.0152(0.87°)
diskon	[0.2026 -0.0831 -0.1405]	[0.4989 0.5485 -0.0513]	0.3393	0.0621(3.56°)
hollbase	[0.2295, -0.0330, -0.2559]	[0.5304, 0.5204, 0.0102]	0.3742	0.0127(0.73°)
tripyrimid	[0.2867, -0.0244, -0.1402]	[0.5100, 0.5400, 0.0030]	0.2564	0.0215(1.23°)
hollstar	[0.2773, -0.1334, 0.0650]	[0.5393, 0.3829, 0.1271]	0.2676	0.1903(10.90°)
multishade	[0.3629, 0.1288, 0.0262]	[0.5293, 0.5111, 0.0123]	0.1900	0.0184(1.06°)

Ge Yu Tianyu Liu Ming Liu Lili Guo
yuge@csu.ac.cn tianyu123@stu.xjtu.edu.cn