Dynamic fusion

大学共同利用機関法人 情報・システム研究機構
国立情報学研究所
National Institute of Informatics

Internship Week 20
Segmented Fusion
5th July 2017

Last meeting

- Previously
  - Length of each volume
  - Local to global transformation
- Plan for today's meeting:
  - Read papers
  - Local to Global transform in CPU
  - Volume transform and projection in CPU
  - Then if it works do it in GPU

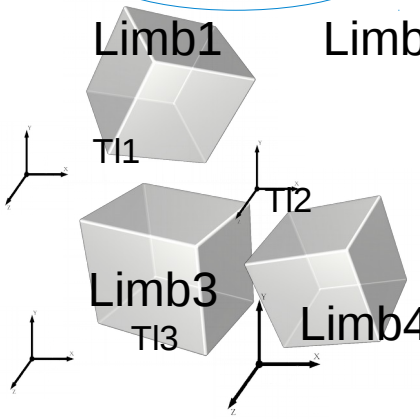
Segmented Fusion Pipeline

For each part

0 Bounding Box (PCA)

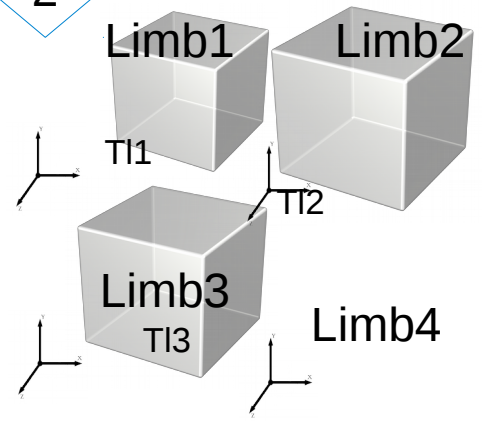
Compute dimensions

1 Create Volumes



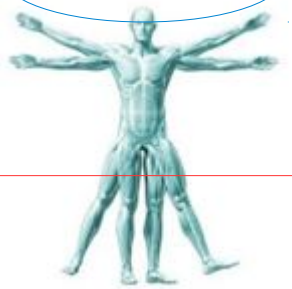
Local to Global coordinates

2



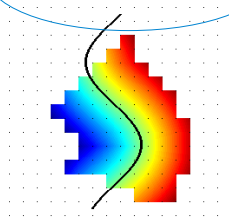
Transform in current pose

3 Align Pose

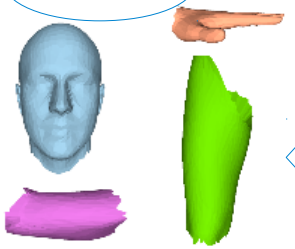


Projection in current image

4 TSDF Fusion

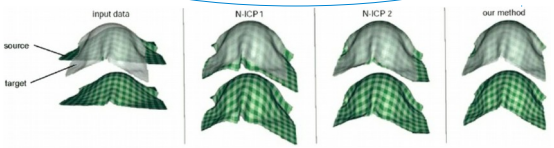


5 Mesh



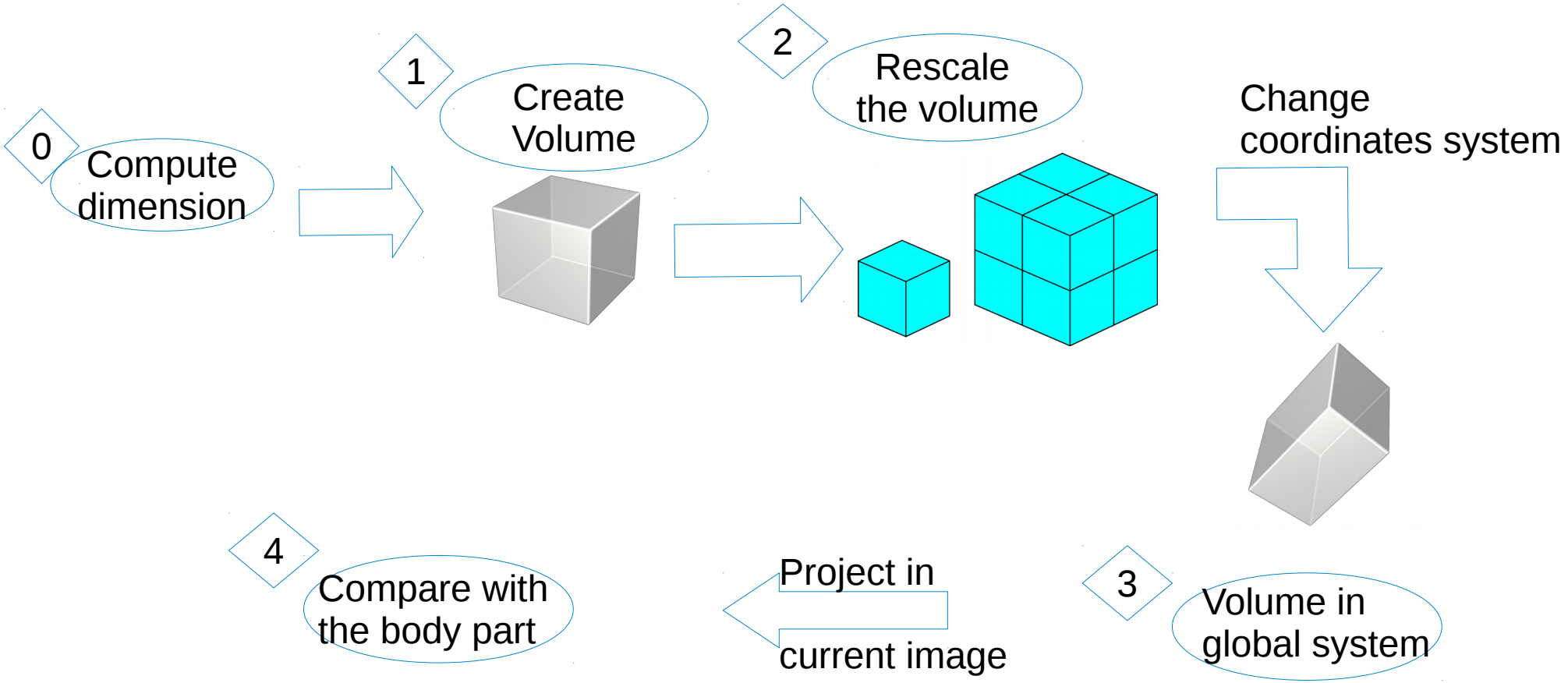
6 Assembling mesh

7 Rigid Alignment



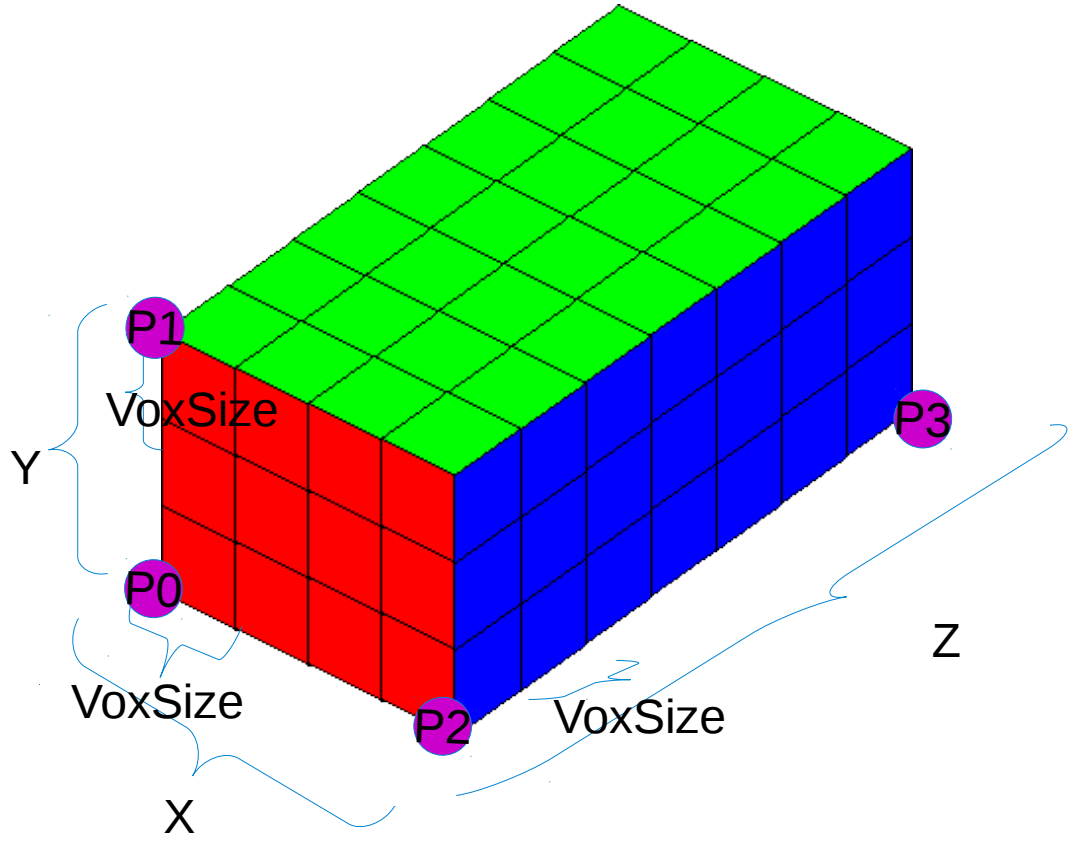
End of 1

Transform Volume:



Volume length depth:

Compute the length of X, Y and Z

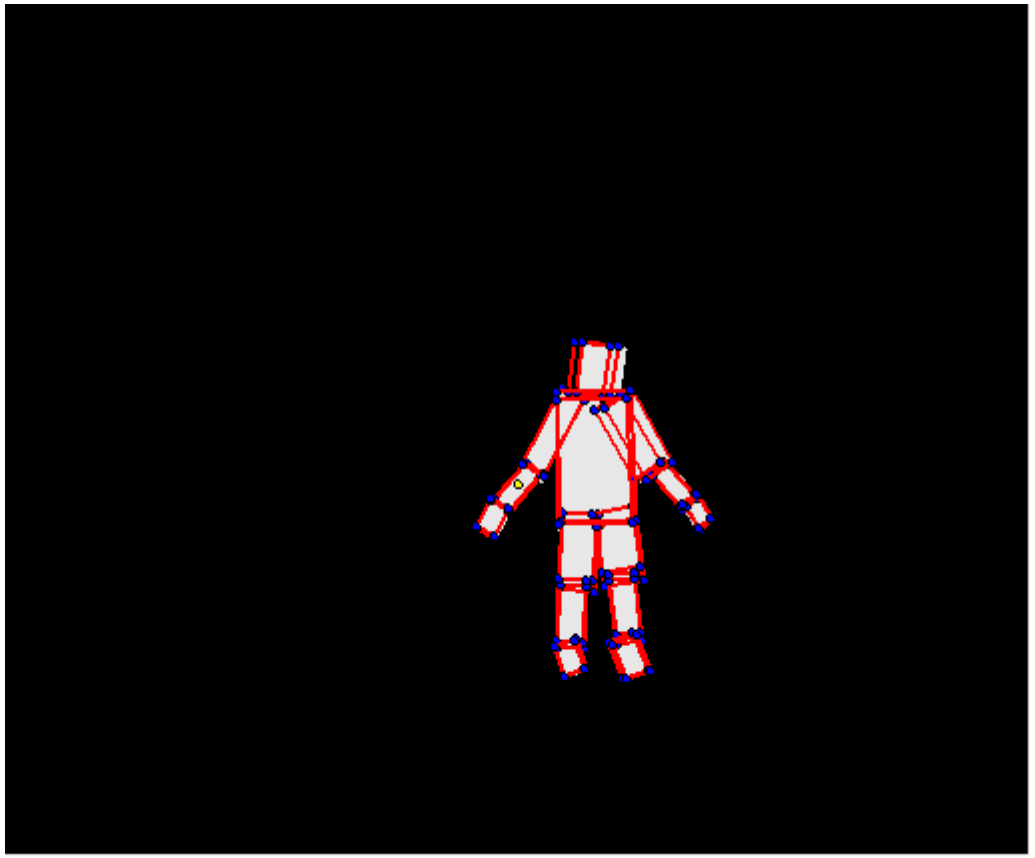


X = norm(P2 – P0)/VoxSize
Y = norm(P1 – P0)/VoxSize
Z = norm(P3 – P2)/VoxSize

Resolution = 1/VoxSize

Overlay bounding Boxes and point of clouds

Fitting the clouds of points in bounding boxes with GPU



Algo :
1) Compute dimension and scaling
2) Get transfo local to global using eigen vector + center of cloud of points in Global coordinates
3) Rescale
4) Transform
5) Project the cloud of point in 2D

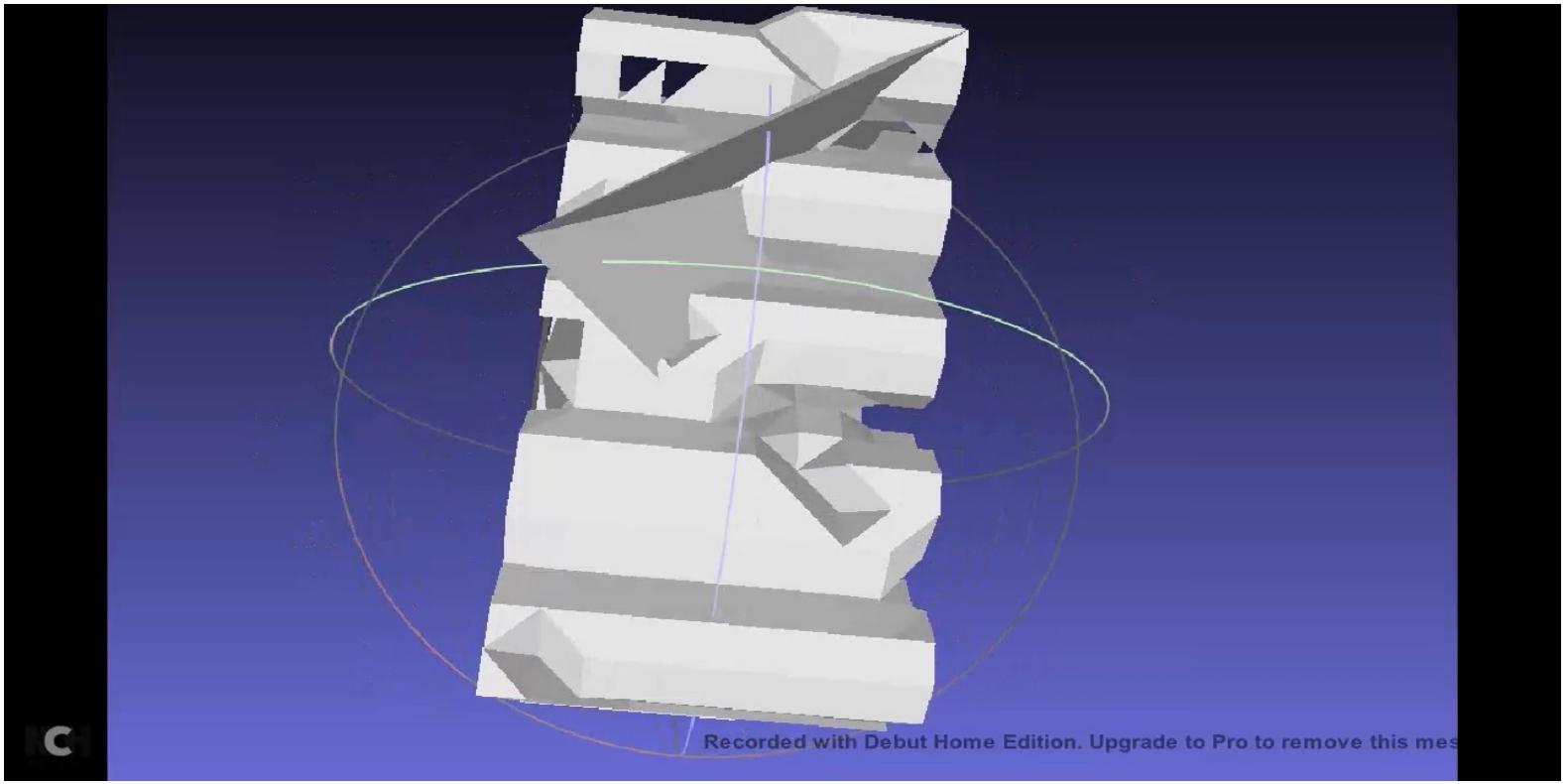
Transfo :

e2	e1	e3	Mean
[[e21	e11	e31	c1]
[e22	e12	e32	c2]
[e23	e13	e33	c3]
[0.	0.	0.	1.]]

Transfo was taken as its transpose as input with GPU

Marching cubes results

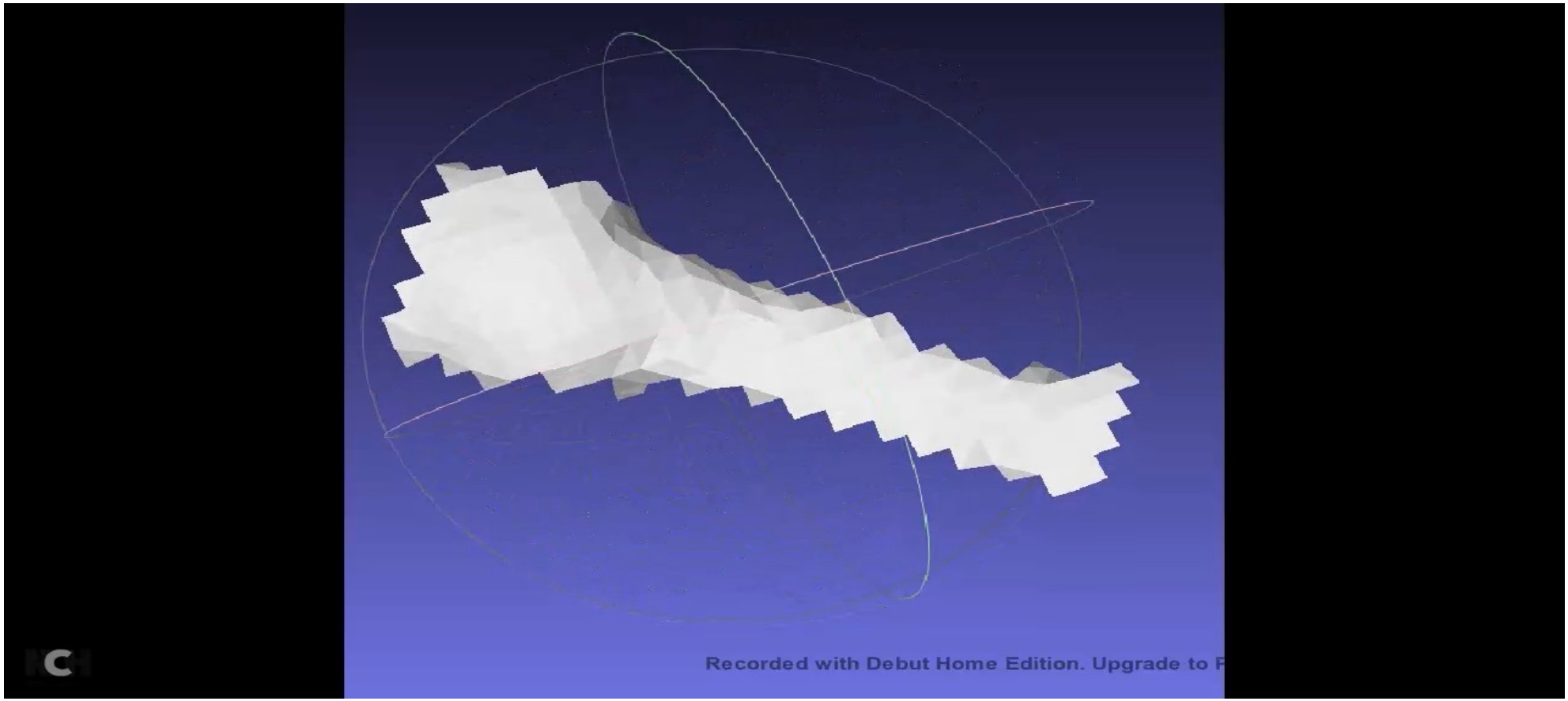
What I have now for the left lower arm



Number of Vertices : about 900

Marching cubes results

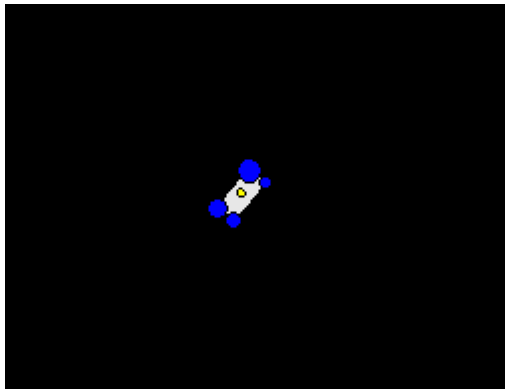
What I want for the left lower arm



Number of Vertices : about 200

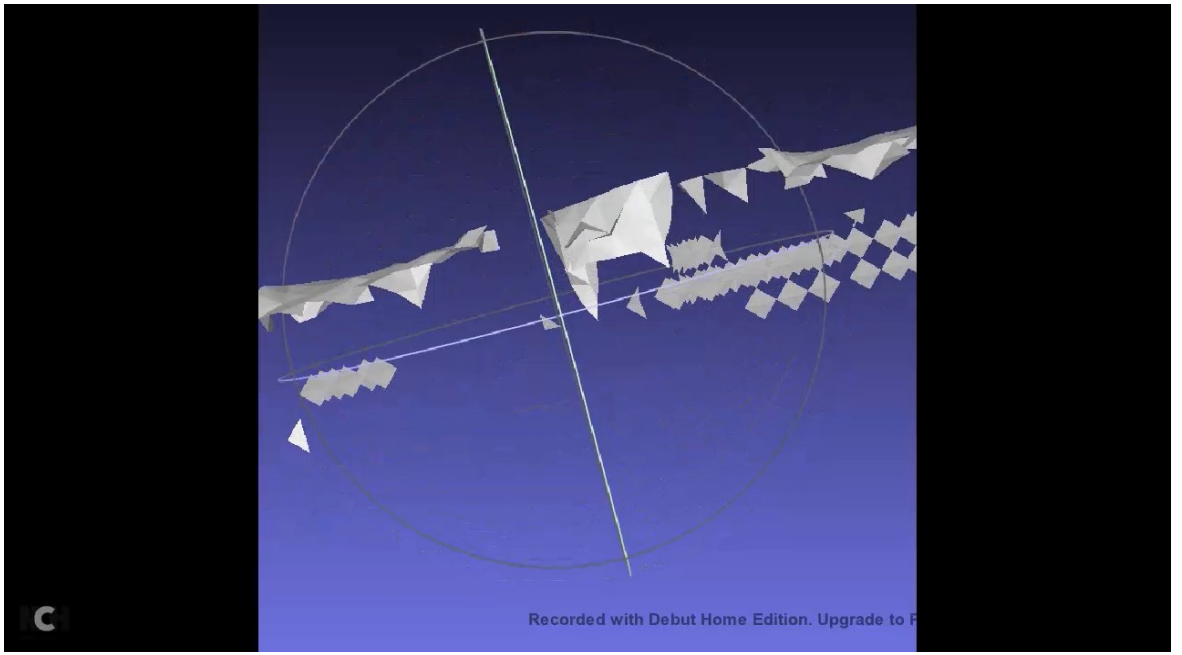
Marching cube debug

Order of point =>
Inversion of X and Y



Count from smallest to biggest

What I used to have



Transfo :

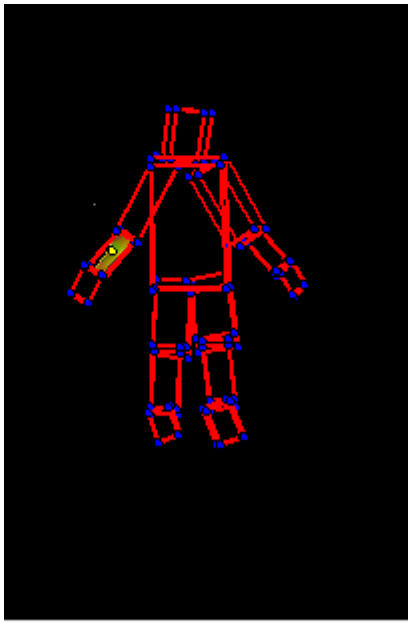
e1	e2	e3	Mean
[[e11	e21	e31	c1]
[e12	e22	e32	c2]
[e13	e23	e33	c3]
[0.	0.	0.	1.]]

Marching cube debug

I used to have different result for GPU in CPU.

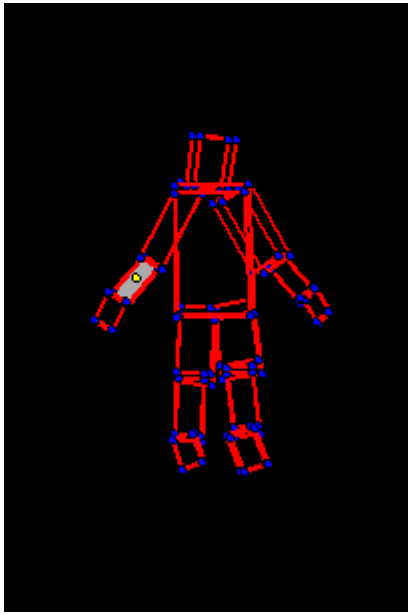
Used Image.Vtx instead of depth_image in CPU.

Number of correspondence



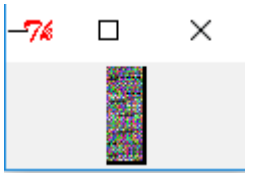
Made with Global Volume

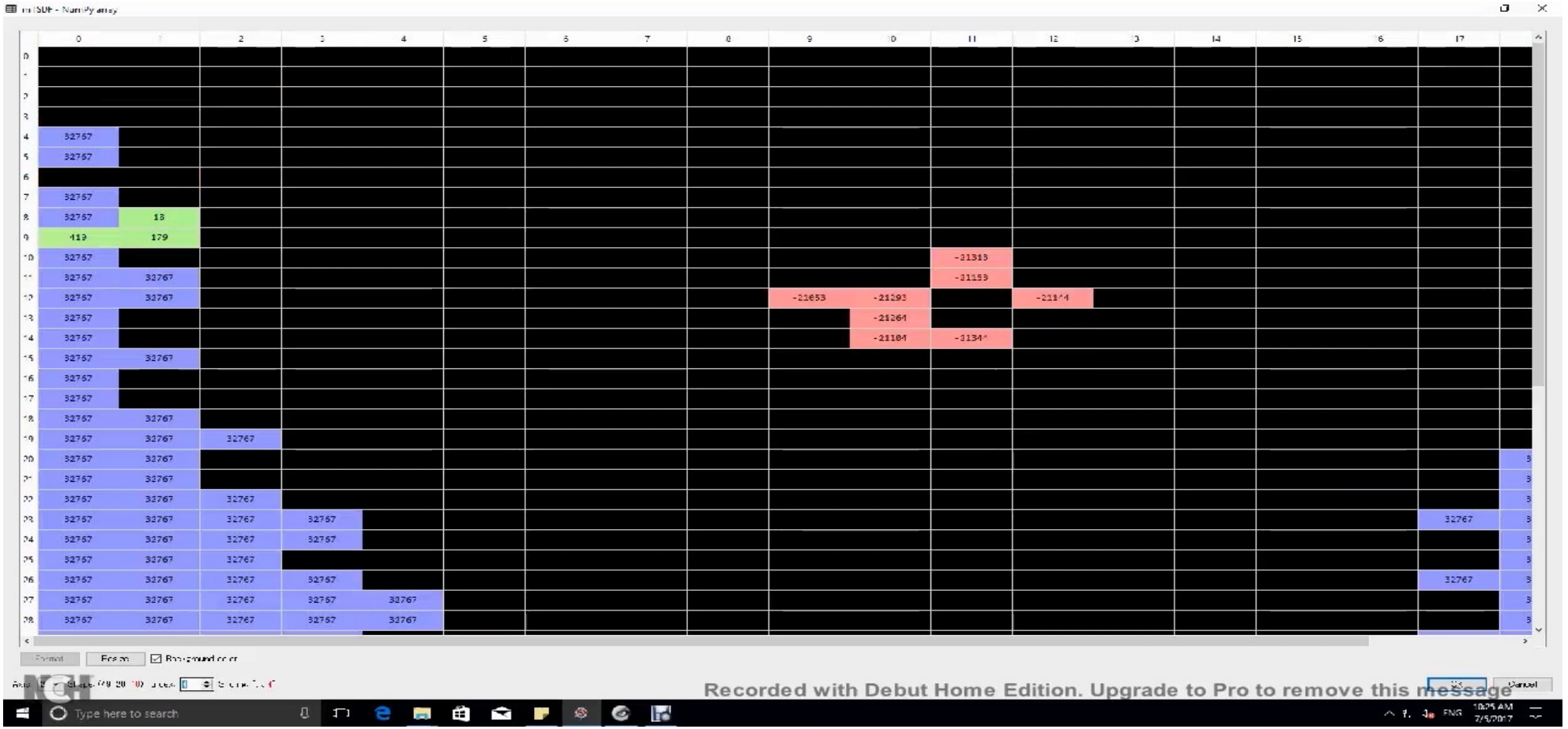
For lower arm left :
9800 Voxels
2400 miss
7400 pass



Made with Local Volume

Marching cube debug





Observed input (TSDF of the lower arm left) given by both CPU and GPU

Stitching

Algo :
1) Define interface between connected body parts (SP)
2) Compute weighted sum of square distance (SP) or Energy between inferface (Killing Fusion)
3) Minimize Energy or distance.

SP may have some issue since the model is not complete.

Action plan

- Reading papers : conceive algo
- Fusion for each segmented body part separately:
  - Local transform
  - Fuse one by one
  - Align globally

Q&A

- Writing report
- GPU code on github?