Custom Fit - B1, B2, B3 (V1)

Summary

The No Patterns Custom-Fit project is broken down into three main bodies: B1, B2, and B3. B1 is the optimized triangle mesh created from point cloud data gathered from any source, but specifically a body scanner. B2 is the wireframe of the piece-wise b-spline body interpolated from the triangle mesh, which is a continuous loop of the points of measure on the body. The points of measure are points on the body that are important measurements to determine the shape of the body. These POMs must be standardized from a body to another body in order for them to be a useful link of comparison between bodies (subject to subject). The wireframe can be controlled by the POMs by a linear transformation to create a frame that represents how the garment should "sit" on the subject's body. B3 is the composition of the extracted contour curves (The blending of the transformed wireframe) and the style curves that are constraints/rules given by the intended design of the finished garment that creates a piece-wise triangle mesh cut into body sections for higher accuracy of the shape of the different regions of the body. This is the Garment Body or "Garbody" because it is effectively the representation of the perfect body that fits inside the garment. To transform the 3D Garbody to a 2D pattern, the Gaussian curvature of the bodies is used for collision detection and tension of the surfaces, which are cut with darts to release the tension and allow 0 curvature, thus flattening the body.

Target End Date

August 30th

North Star

On-demand custom-fit patterns

Squad

EAT Lead - Lily

Objective Owner - Nick

PE Support - Julia

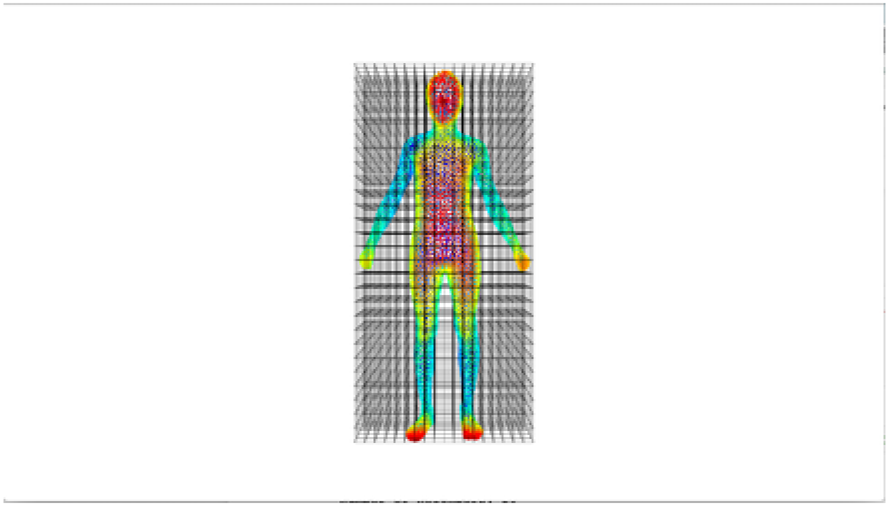
Scope

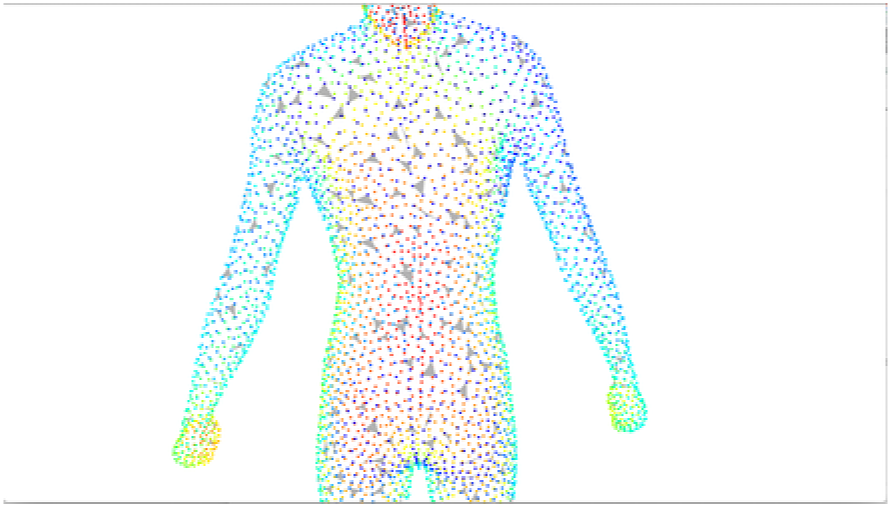
B1

(1) Point Cloud Data > Triangle Mesh > (2) Optimized Triangle Mesh

(1) Point Cloud Triangulation Algorithms

1. Sub mesh Triangulation
    a. Create a box grid of the point cloud
        i. Get the bounding box dimensions of the point cloud
        ii. Get the grid box dimensions (Input number of points per grid box)
            1. For each point in the point cloud
                a. Find n KNN neighbors
                b. Get the bounding box dimensions of the n+1 points
            2. Calculate the median bounding box dimensions
        iii. Create a box grid of the point cloud using the grid box dimensions by creating intervals of (x, y, z) that capture each box the entire bounding box of the point cloud has been captured by the grid boxes



    b. Sub mesh Chain Construction
        i. Initial Triangle
            1. Any point which has never been a component of other chains built from previous chain constructions can be selected as an initial point
            2. Connect the initial point, Point A, with the closest unused point, Point B, to create the initial edge
            3. Find the third point, Point C, to complete the triangle that satisfies these conditions
                a. max(AB, AC, BC) < sum(AB, AC, BC) - max(AB, AC, BC)
                    i. The biggest side of the triangle must be smaller than the sum of the remaining sides
                b. d = sum(AB, AC, BC) - min(AB, AC, BC)
                    i. Finding Point C where d is minimized prioritizes more isosceles-like triangles where AC and BC are minimized while being closer in length to each other (The difference between AC and BC is minimized)
                c. Overlapping Triangles
                    i. Type I Overlap Error – (V1 x AC) * (V1 x V2) > 0 and (V2 x AC) * (V2 x V1) > 0 – Where V1 and V2 are unit vectors that represent the adjacent sides of the previously built triangle
                        1. When at least one side containing the Point C locates between the adjacent sides of the triangle previously created
                    ii. Type II Overlap Error – (AC x V1) * (AC x AB) > 0 and (AB x V1) * (AB x AC) > 0 and (AC x V2) * (AC x AB) > 0 and (AB x V2) * (AB x AC) > 0 – Where V1 and V2 are unit vectors that represent the adjacent sides of the previously built triangle
                        1. When the angle of the initial triangle at the Point C contains the angle of another already existing triangle where Point C is also the vertex of this triangle
            4. 

ii. Expanding Vectors

1. After the initial triangle is found, it is used as the seed for the chain construction of more triangles

2. e = ((V * W) * V - W) / LENGTH((V * W) * V - W) – Where V is the unit vector in the direction of the side for which the expanding vector is being calculated and W is the vector of the other side sharing the same vertex

iii. Subsequent Construction

1. Triangle Construction Conditions

a. The Vector from the considered edge to the third point and the expanding vector of the edge are aligned on the same side of the edge

i. Consider the edge AB. The vectors from the edge AB to various points must be aligned on the same side of the edge that the expanding vector is facing otherwise they are not considered as potential third points

ii. To check this condition let u be a vector from the edge AB to a potential point. If u * e > 0 then the angle between vector u and the expanding vector is acute, which means they are aligned on the same side of AB and can now be considered as a potential point

b. The triangle containing the third point as the vertex must have angles at the base between 5 and 120 degrees (This range can be changed as an input)

c. The two new edges produced by the third point are not allowed if at least one of these two edges is the common side of an edge already connected to two other triangles

d. The third point must satisfy all the conditions as required for Point C in the construction of the initial triangle. In addition, the Type I and Type II Overlap Errors must be checked at every corner of the triangle instead of only at the corner of the third point

2. After each new triangle is constructed the following attributes are stored

a. The number of triangles each vertex contains

b. The number of triangles each edge contains

c. The expanding vector of each edge

d. The angle of each corner of the triangle

iv. Collection of chains

1. The chain is complete once no new triangles are created and the chain construction procedure repeats with a new initial triangle as a seed until no initial triangles can be found

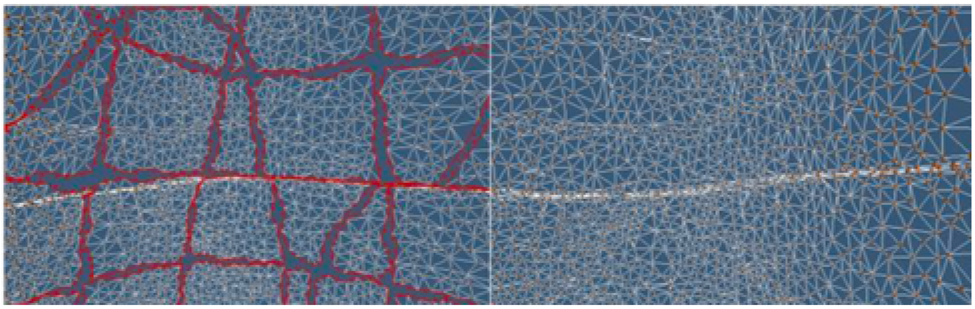
2. The box sub mesh is complete once all the holes are filled

c. Sub mesh integration

i. The Submesh Chain Construction creates a sub mesh for every non-empty box in the grid

ii. Starting with an arbitrary box, find all of the single edges in the sub mesh of the box and connect to the adjacent box sub meshes using the same subsequent construction procedure by having each single edge search for a third point in the adjacent boxes to form additional triangles

iii. Complete the Triangle Mesh once all sub meshes are integrated by filling all of the holes



(2) Optimizing Triangle Mesh Algorithms

1. MARBLE

a. Triangle Centroid Smoothing

B2

Optimized Triangle Mesh > (1) Piece-Wise B-Spline Interpolated Mesh > (2) Wireframe > (3) Transformed Wireframe

(1) Piece-Wise B-Spline Interpolation Algorithms

(2) Wireframe Algorithms

(3) Transforming Wireframe Algorithms

B3

Future Enhancements

Have ideas that can't make this 6 week cycle. List the ideas here so they do not get lost!

Meeting Notes

ADD MEETING DATE HERE ...

Insert Meeting Notes

Additional Resources

Helpful Resources

No Patterns Base: https://airtable.com/tblTFJIJc5n5HJdRb/viwxcJyuml5EtYXXc?blocks=hide

Sub Mesh Triangulation: Method_of_3D_mesh_reconstruction_from_point_cloud_.pdf

Triangle Overlap Visualization: https://www.geogebra.org/classic/vzhp4ths

Child Pages