Fall 2017

Unimaginable Form Semantic Exploration in Digital Turn 2.0

Yang Wang

Syracuse University

UNIMAGINABLE FORM
Semantic Exploration in Digital Turn 2.0

YANG WANG

Advisor
Amber Bartosh

"We live in a universe whose age we can't quite compute, surrounded by stars whose distances we don't altogether know, filled with matter we can't identify, operating in conformance with physical laws whose properties we don't truly understand."

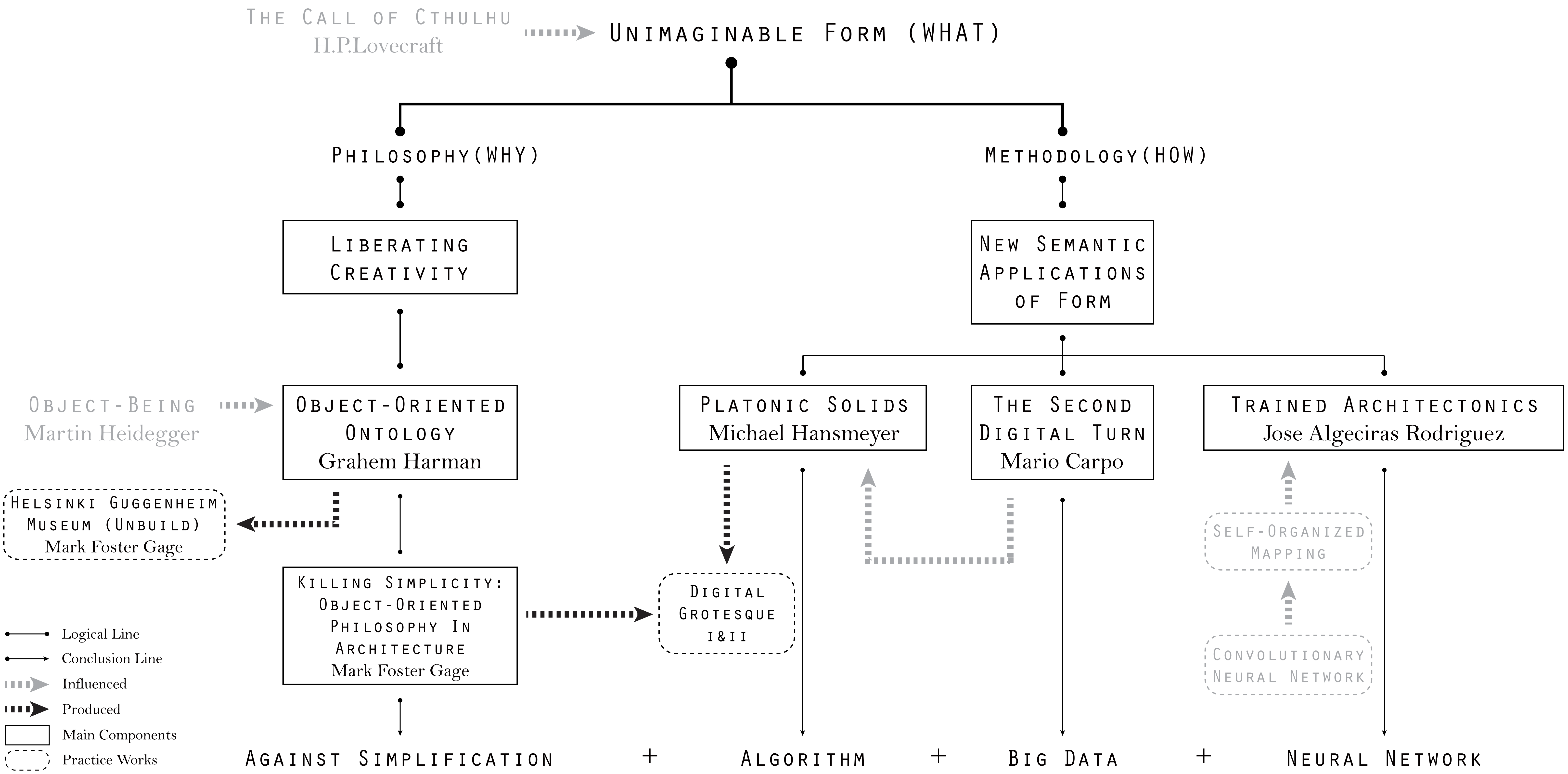
---Bill Bryson, "A Short History of Nearly Everything"

UNIMAGINABLE FORM

"...no architecture known to man or to human imagination, with vast aggregations of night-black masonry embodying monstrous perversions of geometrical laws."

---H. P. Lovecraft, *The Call of Cthulhu*, 1928

Stephan McLeroy
The Mountain of Madness
2015



The Call of Cthulhu
H.P.Lovecraft

Unimaginable Form (WHAT)

Philosophy(WHY)

Methodology(HOW)

Liberating Creativity

New Semantic Applications of Form

Object-Being
Martin Heidegger

Object-Oriented Ontology
Grahem Harman

Platonic Solids
Michael Hansmeyer

The Second Digital Turn
Mario Carpo

Trained Architectonics
Jose Algeciras Rodriguez

Helsinki Guggenheim Museum (Unbuild)
Mark Foster Gage

Killing Simplicity: Object-Oriented Philosophy In Architecture
Mark Foster Gage

Digital Grotesque I&II

Self-Organized Mapping

Convolutionary Neural Network

Logical Line
Conclusion Line
Influenced
Produced
Main Components
Practice Works

Against Simplification + Algorithm + Big Data + Neural Network

Using Big Data-Driven algorithm based on the Self-Organized Mapping(SOM) and Convolutionary Neural Network (CNN) to do the Computational Generative Form Design with huge complexities of Form Reality.

Philosophy
(WHY)

Librating Creativity

"Focusing on the vast withdrawn complexities of an architectural project would not only be a welcome antidote to the trope of inventing architectural concepts and diagramming them for easy comprehension, but would also liberate architects' creativity."

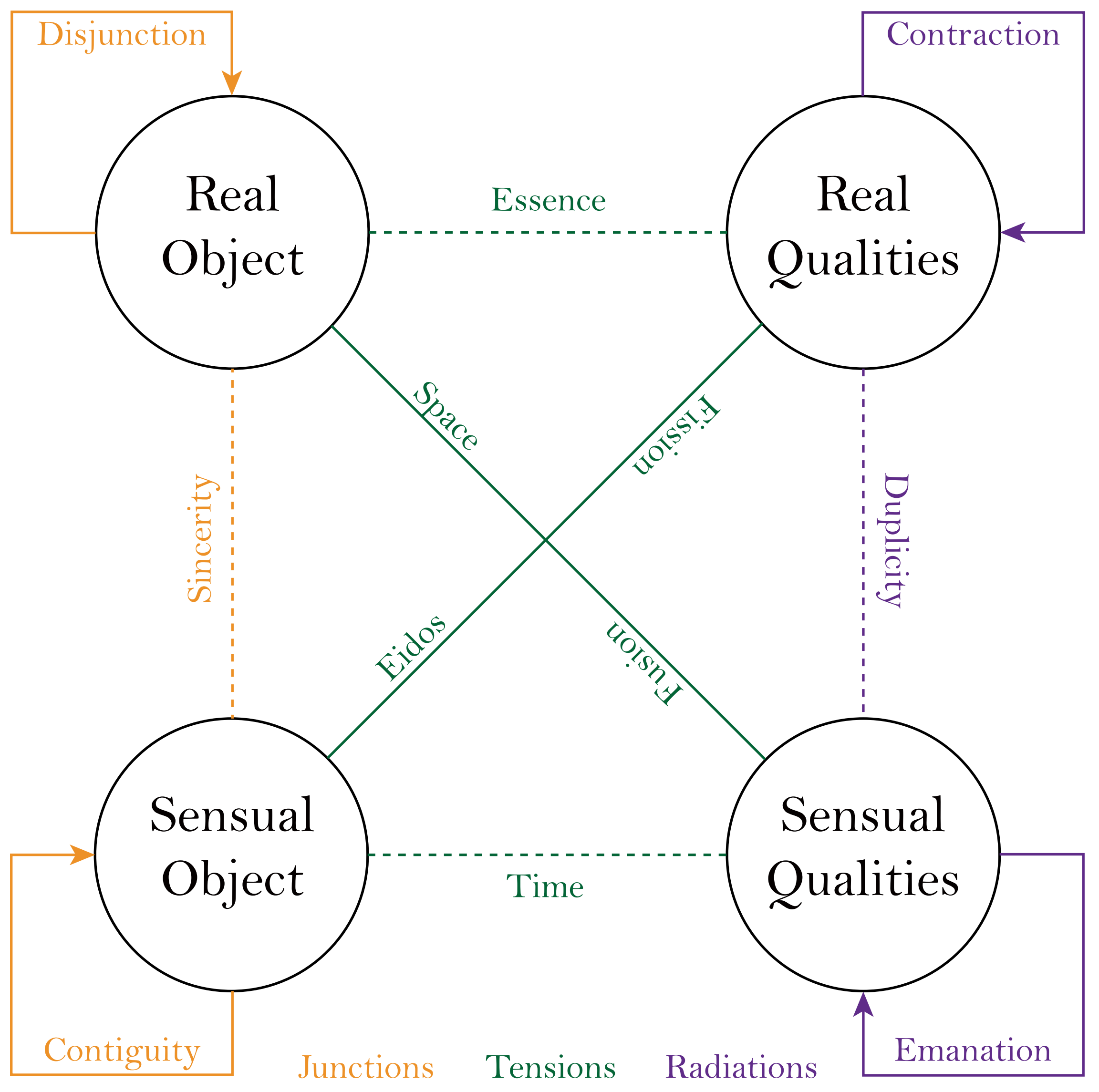
---Mark Foster Gage, *Killing Simplicity*, 2016

The initiatively avoidance on the complexities of form has imprisoned architects' creativity.

Object-Oriented Ontology

“The only way to do justice to objects is to consider that their reality is free of all relation, deeper than a reciprocity.”

---Graham Harman, *The Quadruple Object*, 2011



Computational

"After decades of computational calculation, exactitude, and the translation of information and diagrams into mostly banal, literal buildings, perhaps inference through illusion and innuendo offers fertile fields for developing newer, slipperier, and more uncertain forms of architectural practice."

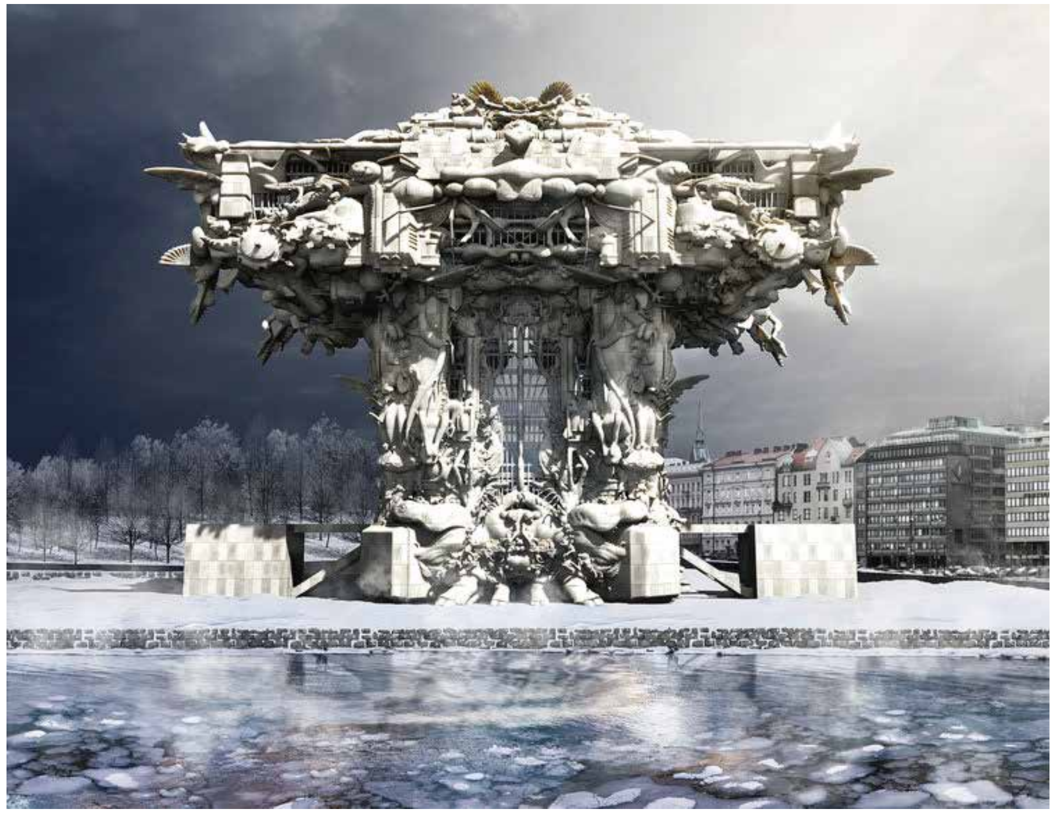
---Mark Foster Gage, *Killing Simplicity*, 2016

High Density

"In Object- oriented ontology, real objects are simply not fully knowable. This is not a mystical notion but rather one that emerges from the sheer infinitude of qualities and relations-as-objects that define an object. As such, it is an information-dense proposition."

---Mark Foster Gage, *Killing Simplicity*, 2016

(Representing the density of form information)



Mark Foster Gage
Helsiki Guggenheim Museum (Unbuilt)
2014

Mark Foster Gage
Helsiki Guggenheim Museum (Detail)
2014

Methodology
(HOW)

Digital Turn 2.0

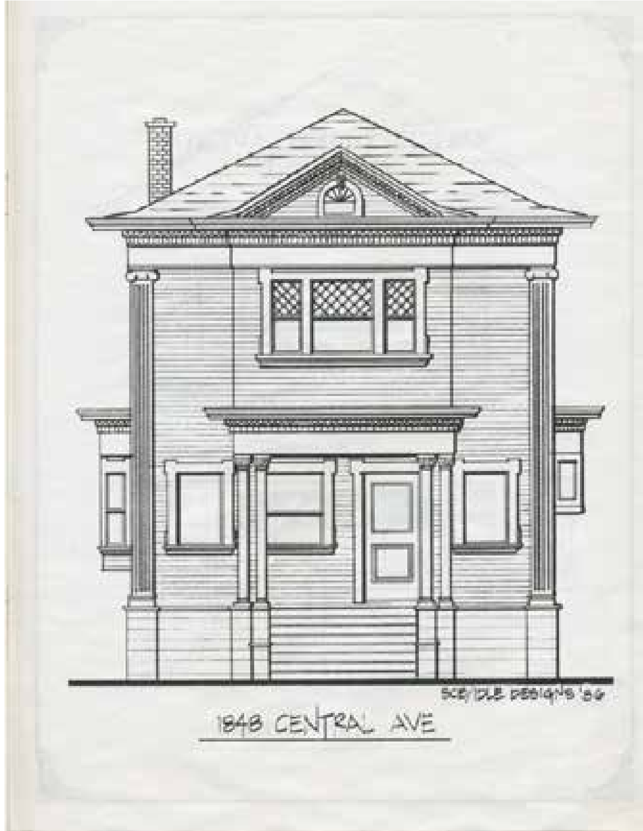
Big Data - Driven

"...designers use 'big data' to notate reality as it appears at any chosen scale, without having to convert it into simplified and scalable mathematical notations or laws."

---Mario Carpo, Breaking the Curve: Big Data and Design, 2017

Pre - Digital Turn

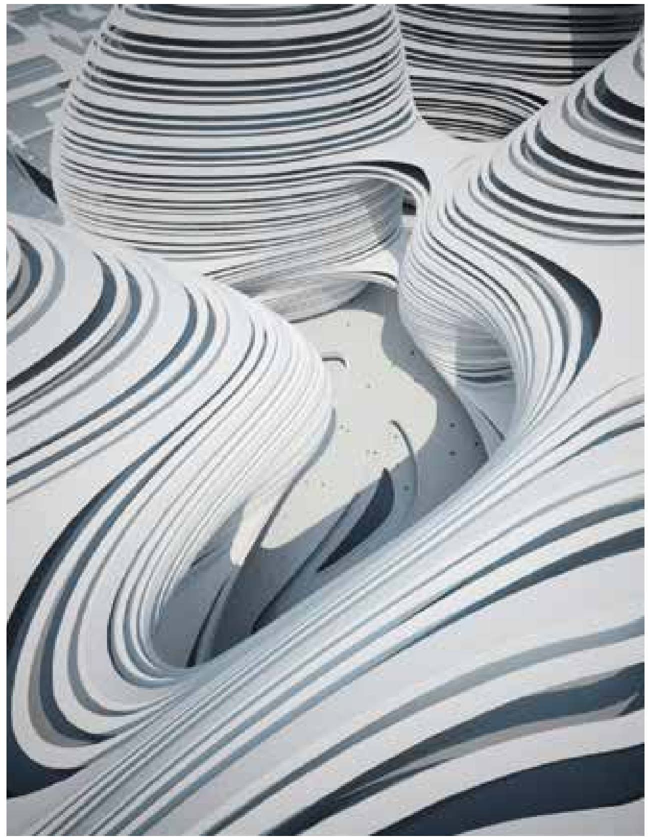
Ruler, Pencil, Eraser, Compass



Stephan DeLacey Idle & Scott C. Brady
Victorian Coloring Book
1987

Digital Turn 1.0

Spline Module



Zaha Hadid
Soho Galaxy
2007

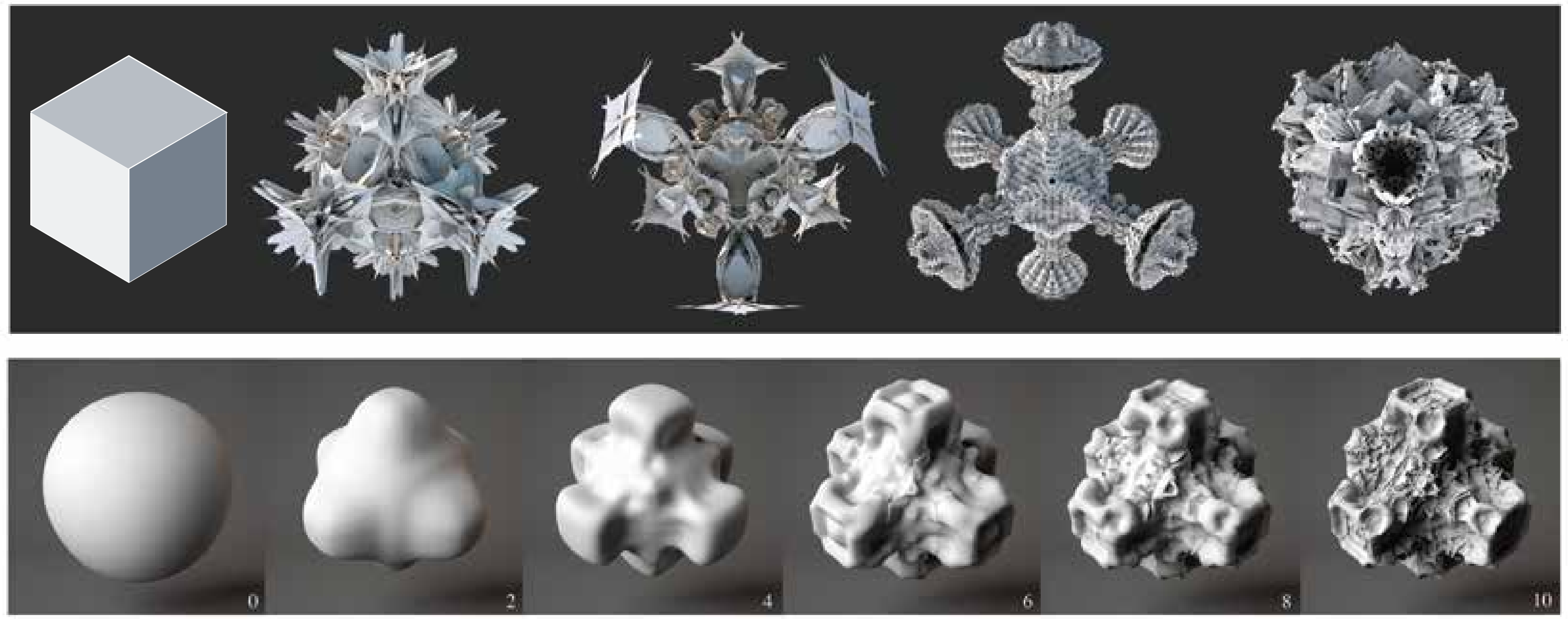
Algorithm

“Similar processes do not necessarily beget similar shapes. Understanding these processes, on contrary, will help us shape better things.”

---Mario Carpo, THE ALPHABET AND THE ALGORITHM, 2011

“we didn’t design the form, we designed the process that generated the form.”

---Michael Hansmeyer, TED lecture, 2015

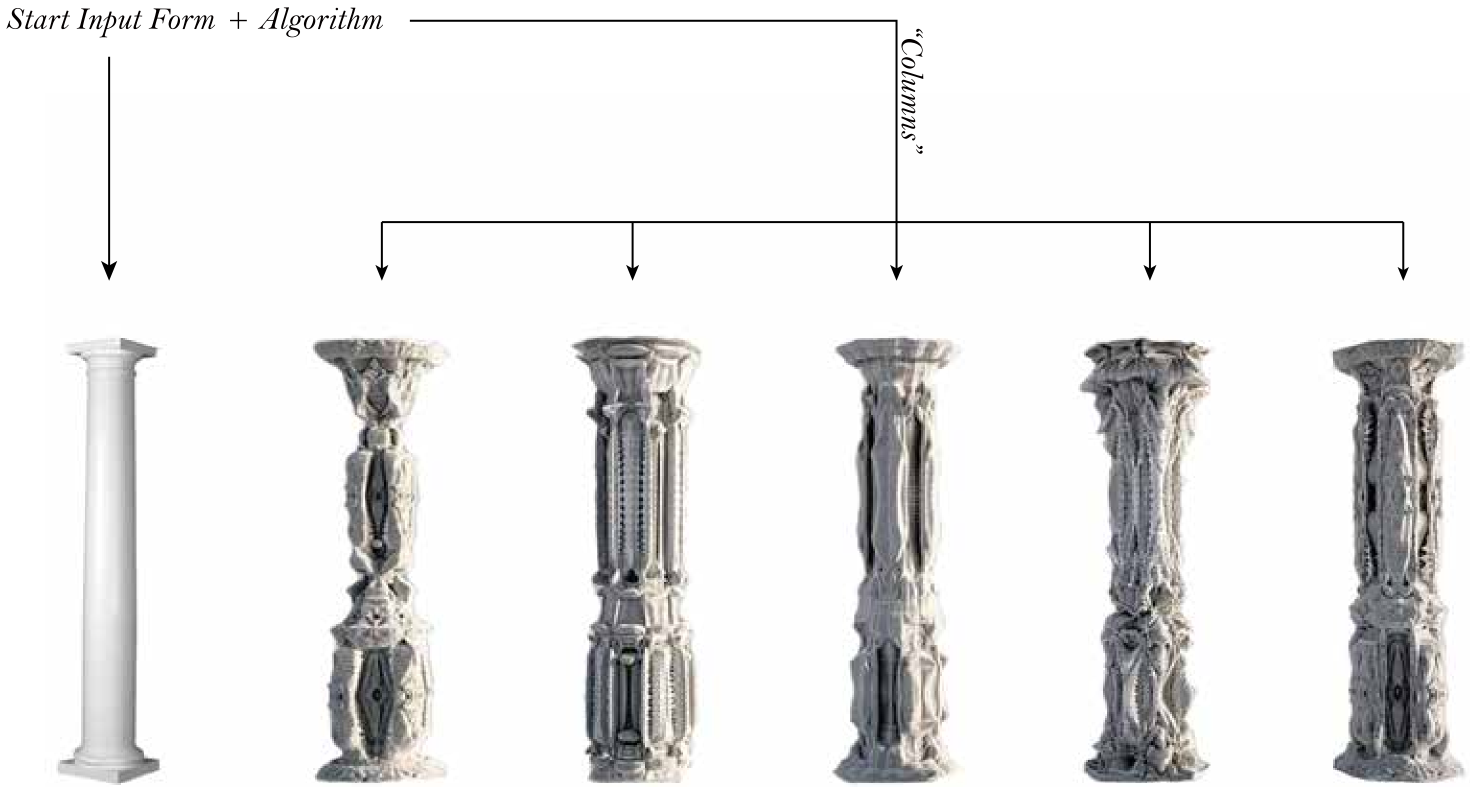


Michael Hansmeyer
Platonic Solids
2008

Learning from architectural forms

“...a far effective way to create forms is to use information that is already contained in forms...”

---Michael Hansmeyer, TED lecture, 2015

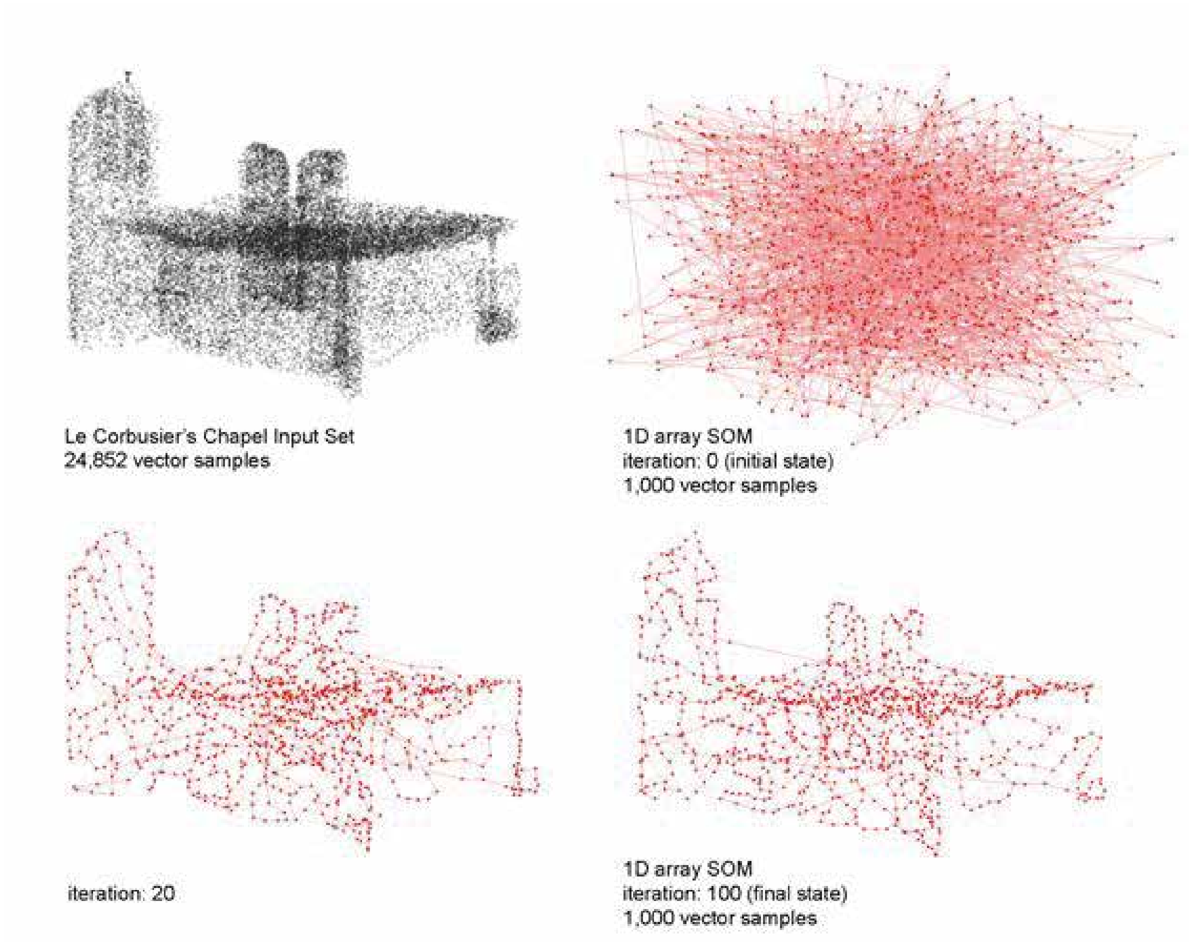


Michael Hansmeyer
Columns
2010

Neural Network

"The role of the designer here is the role of an instructor of learning machines."

---Jose Algeciras Rodriquez, *Trained Architectonics*, 2016



Jose Algeciras Rodriguez
Trained Architectonics
2016

Algorithm Explanation

Vegetable-Fruit Juice = a*APPLE + b*BANANA + c*CARROT + ...+ n*ANY

(a, b, c, ..., n is the numbers of spoon)

$$f(x) = w_1(A)+w_2(B)+w_3(C)+ \cdots +w_n(N)=\int_1^n w_n(N)$$

w - Weight of Iteration Size

x - Initial Learning Sample

N - Forms of Learning Object

1) Setting up the existing 3D models transmitted into point-cloud

2) Using SOM to process the dimensionality reduction which could simplify three dimensional sample set into two dimensional classification algorithm

3) Using CNN to sovle the classification function

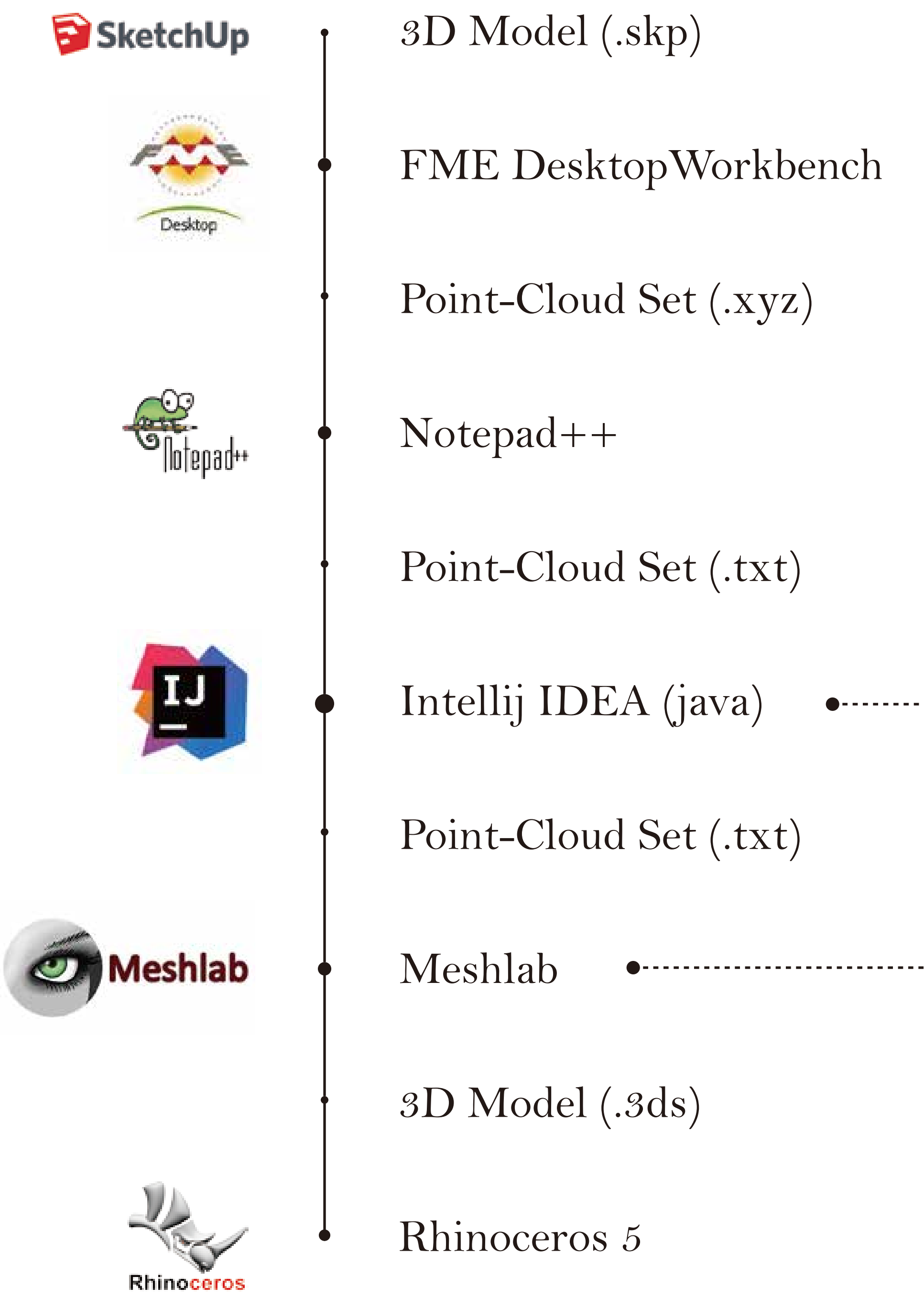
4) Control and adjust the times of algorithm iteration

5) Setting the next learning sample model ready

6) Repeat the step 2) to step 4) untill finishing all models

7) Export the final point-cloud set into Grasshopper to do the data processing and rendering

First Prototype Description



3D Model (.skp)

FME DesktopWorkbench

Point-Cloud Set (.xyz)

Notepad++

Point-Cloud Set (.txt)

Intellij IDEA (java)

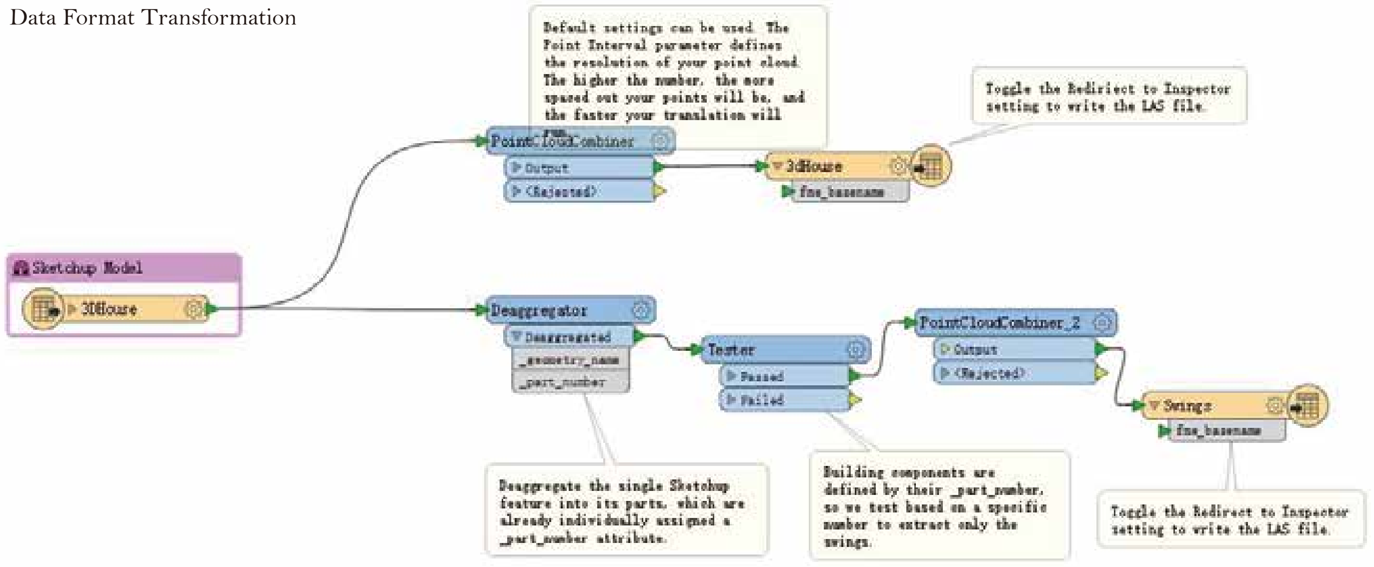
Point-Cloud Set (.txt)

Meshlab

3D Model (.3ds)

Rhinoceros 5

Data Format Transformation



Default settings can be used. The Point Interval parameter defines the resolution of your point cloud. The higher the number, the more spaced out your points will be, and the faster your translation will run.

Toggle the Redirect to Inspector setting to write the LAS file.

Deaggregate the single Sketchup feature into its parts, which are already individually assigned a _part_number attribute.

Building components are defined by their _part_number, so we test based on a specific number to extract only the swings.

Toggle the Redirect to Inspector setting to write the LAS file.

1. Importing .txt file into Meshlab

2. Surface Reconstruction based on Ball Pivoting

3. Smooth Face Normals

4. Exporting .3ds format File

Core Script Algorithm

1. Untrained Random Point-Cloud Set: Volume 10,000

2. Writting Untrained Points into Array of 100 by 100

3. Reading the Point-Cloud of Learning Sample

4. Calculating the Euclidean Distance of Untrained Points and Learning Sample One by One

5. Filtering Array[i][j] based on the radius of influence

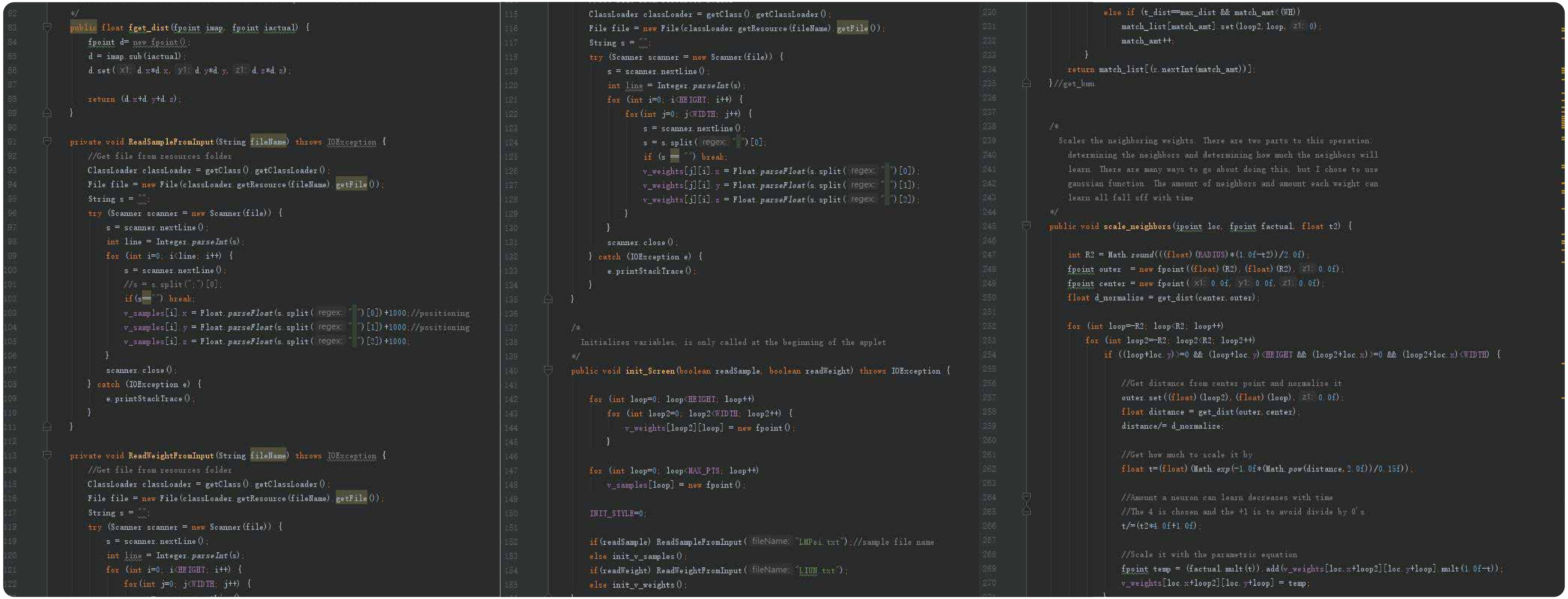
6. Return Results to main script

7. Moving Selected Untrained Points with the square foot of distance

8. Isolating the matched points of Learning Sample

9. Repeating setting iteration times from step 4 to step 8

10. Reading Output and Exporting .txt format File

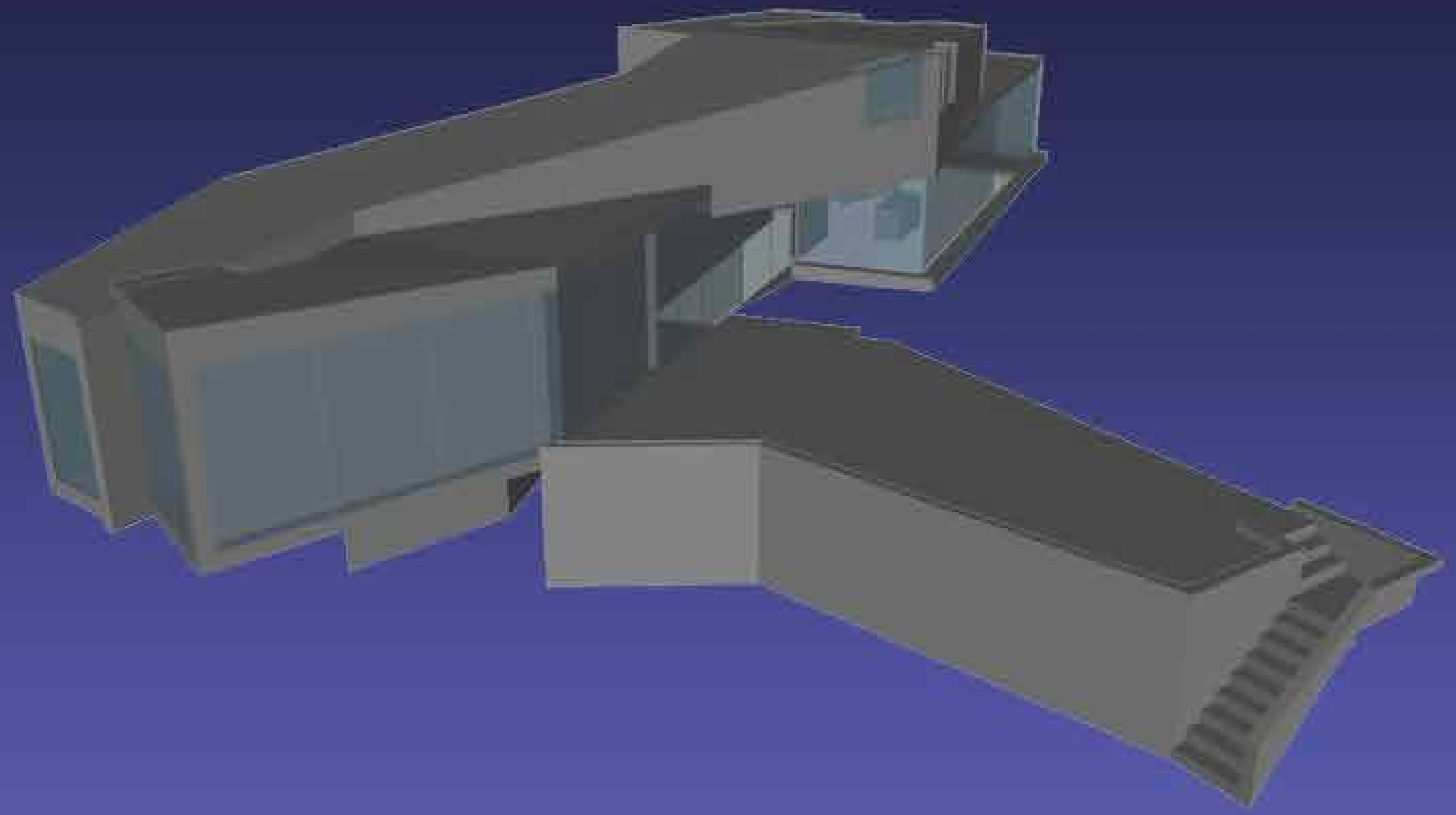


```
    */
    public float fget_dist(fpoint imap, fpoint iactual) {
        fpoint d= new fpoint();
        d = imap.sub(iactual);
        d.set( x1: d.x*d.x, y1: d.y*d.y, z1: d.z*d.z);

        return (d.x+d.y+d.z);
    }

    private void ReadSampleFromInput(String fileName) throws IOException {
        //Get file from resources folder
        ClassLoader classLoader = getClass().getClassLoader();
        File file = new File(classLoader.getResource(fileName).getFile());
        String s = "";
        try (Scanner scanner = new Scanner(file)) {
            s = scanner.nextLine();
            int line = Integer.parseInt(s);
            for (int i=0; i<line; i++) {
                s = scanner.nextLine();
                //s = s.split(";")[0];
                if(s=="") break;
                v_samples[i].x = Float.parseFloat(s.split( regex: " ")[0])+1000;//positioning
                v_samples[i].y = Float.parseFloat(s.split( regex: " ")[1])+1000;//positioning
                v_samples[i].z = Float.parseFloat(s.split( regex: " ")[2])+1000;
            }
            scanner.close();
        } catch (IOException e) {
            e.printStackTrace();
        }
    }

    private void ReadWeightFromInput(String fileName) throws IOException {
        //Get file from resources folder
        ClassLoader classLoader = getClass().getClassLoader();
        File file = new File(classLoader.getResource(fileName).getFile());
        String s = "";
        try (Scanner scanner = new Scanner(file)) {
            s = scanner.nextLine();
            int line = Integer.parseInt(s);
            for (int i=0; i<HEIGHT; i++) {
                for(int j=0; j<WIDTH; j++) {
```

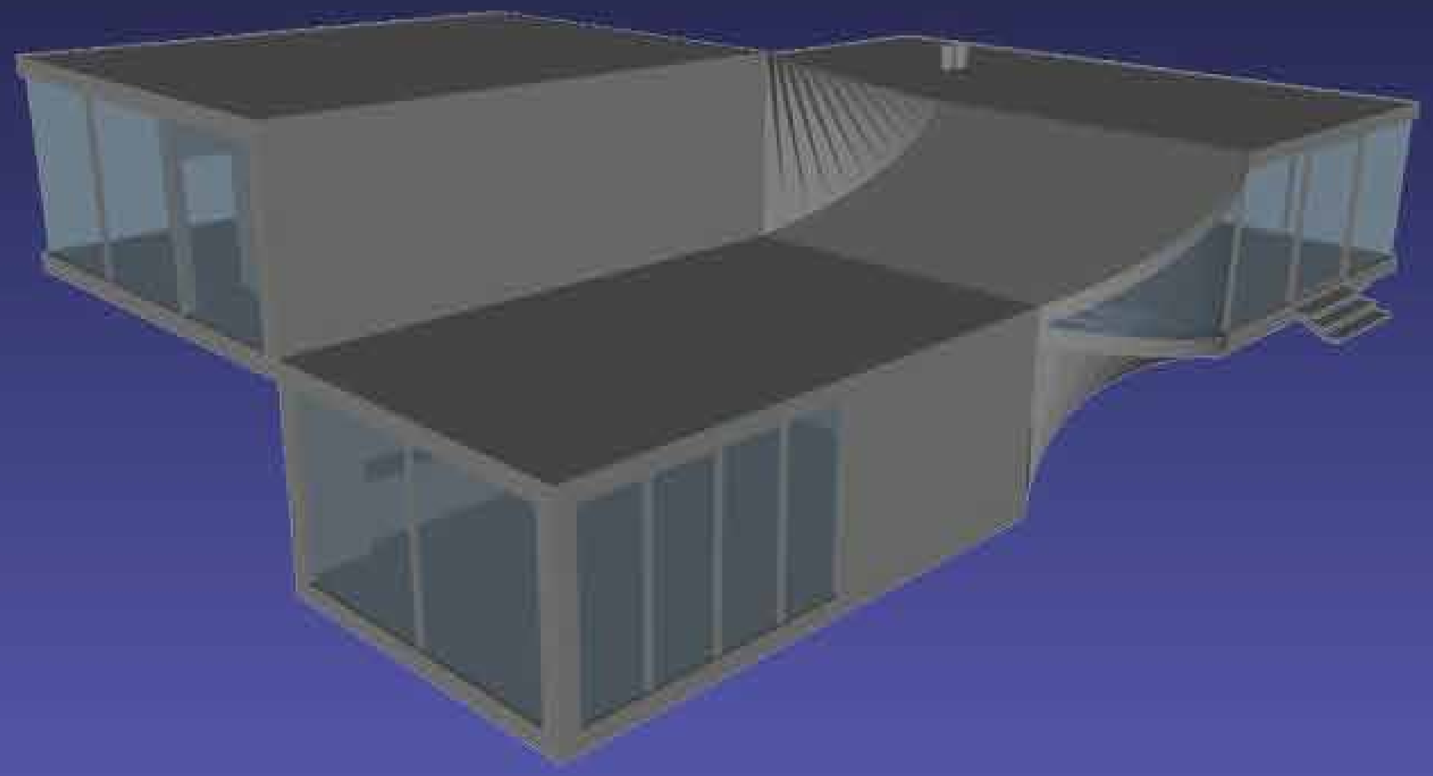
```
        ClassLoader classLoader = getClass().getClassLoader();
        File file = new File(classLoader.getResource(fileName).getFile());
        String s = "";
        try (Scanner scanner = new Scanner(file)) {
            s = scanner.nextLine();
            int line = Integer.parseInt(s);
            for (int i=0; i<HEIGHT; i++) {
                for(int j=0; j<WIDTH; j++) {
                    s = scanner.nextLine();
                    s = s.split( regex: ";")[0];
                    if (s == "") break;
                    v_weights[j][i].x = Float.parseFloat(s.split( regex: " ")[0]);
                    v_weights[j][i].y = Float.parseFloat(s.split( regex: " ")[1]);
                    v_weights[j][i].z = Float.parseFloat(s.split( regex: " ")[2]);
                }
            }
            scanner.close();
        } catch (IOException e) {
            e.printStackTrace();
        }
    }

    /*
      Initializes variables, is only called at the beginning of the applet
    */
    public void init_Screen(boolean readSample, boolean readWeight) throws IOException {

        for (int loop=0; loop<HEIGHT; loop++)
            for (int loop2=0; loop2<WIDTH; loop2++) {
                v_weights[loop2][loop] = new fpoint();
            }

        for (int loop=0; loop<MAX_PTS; loop++)
            v_samples[loop] = new fpoint();

        INIT_STYLE=0;

        if(readSample) ReadSampleFromInput( fileName: "LHPei.txt");//sample file name
        else init_v_samples();
        if(readWeight) ReadWeightFromInput( fileName: "LIUN.txt");
        else init_v_weights();
```

```
                else if (t_dist==max_dist && match_amt<(WH))
                    match_list[match_amt].set(loop2, loop, z1: 0);
                    match_amt++;
            }
        return match_list[(r.nextInt(match_amt))];
    }//get_bmu


    /*
      Scales the neighboring weights. There are two parts to this operation:
        determining the neighbors and determining how much the neighbors will
        learn. There are many ways to go about doing this, but I chose to use
        gaussian function. The amount of neighbors and amount each weight can
        learn all fall off with time
    */
    public void scale_neighbors(ipoint loc, fpoint factual, float t2) {

        int R2 = Math.round(((float)(RADIUS)*(1.0f-t2))/2.0f);
        fpoint outer  = new fpoint((float)(R2),(float)(R2), z1: 0.0f);
        fpoint center = new fpoint( x1: 0.0f, y1: 0.0f, z1: 0.0f);
        float d_normalize = get_dist(center, outer);

        for (int loop=-R2; loop<R2; loop++)
            for (int loop2=-R2; loop2<R2; loop2++)
                if ((loop+loc.y)>=0 && (loop+loc.y)<HEIGHT && (loop2+loc.x)>=0 && (loop2+loc.x)<WIDTH) {

                    //Get distance from center point and normalise it
                    outer.set((float)(loop2),(float)(loop), z1: 0.0f);
                    float distance = get_dist(outer, center);
                    distance/= d_normalize;

                    //Get how much to scale it by
                    float t=(float)(Math.exp(-1.0f*(Math.pow(distance,2.0f))/0.15f));

                    //Amount a neuron can learn decreases with time
                    //The 4 is chosen and the +1 is to avoid divide by 0's
                    t/=(t2*4.0f+1.0f);

                    //Scale it with the parametric equation
                    fpoint temp = (factual.mult(t)).add(v_weights[loc.x+loop2][loc.y+loop].mult(1.0f-t));
                    v_weights[loc.x+loop2][loc.y+loop] = temp;
```

FIRST LEARNING SAMPLES



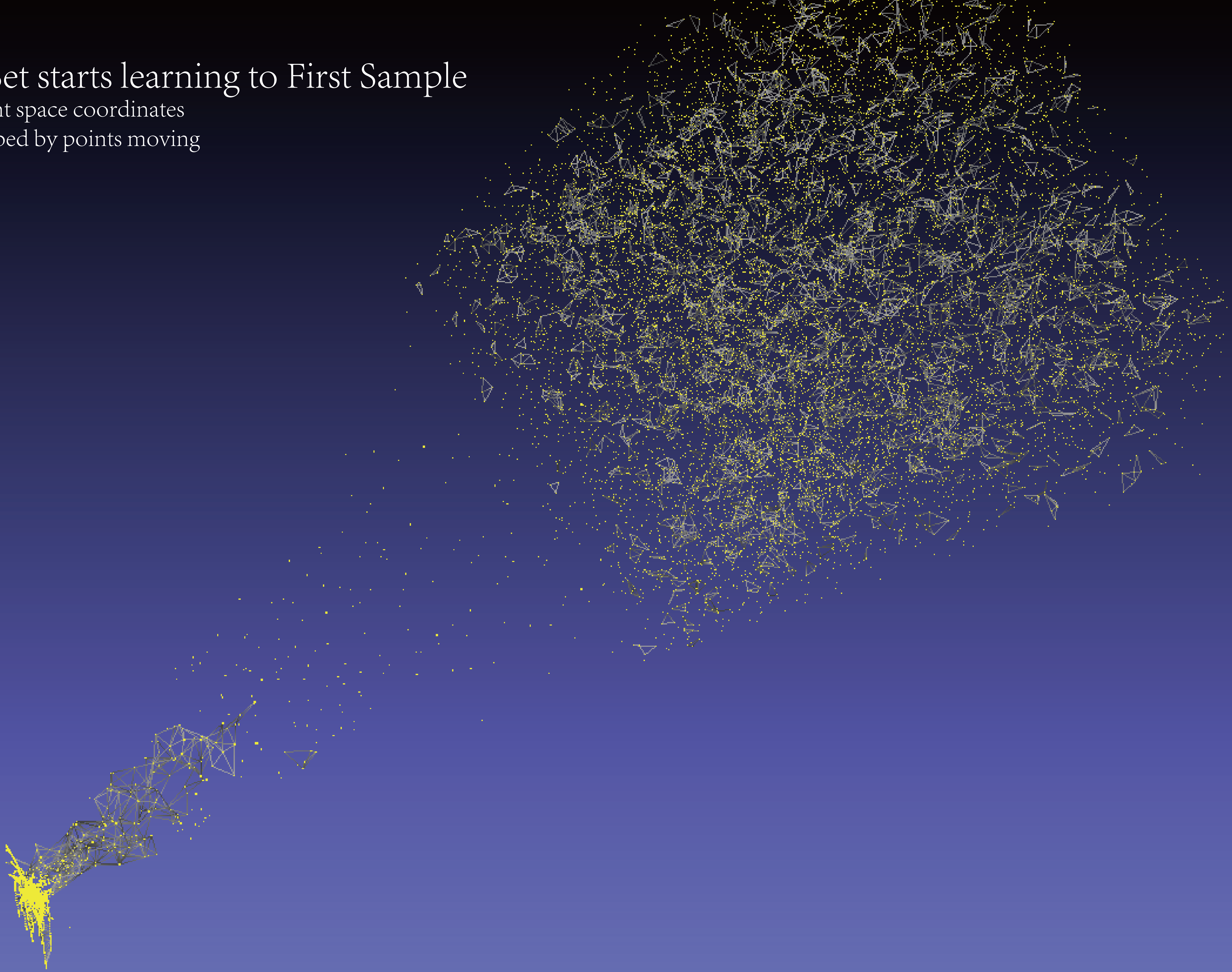
SECOND LEARNING SAMPLES



Untrained Point-Cloud Set starts learning to First Sample

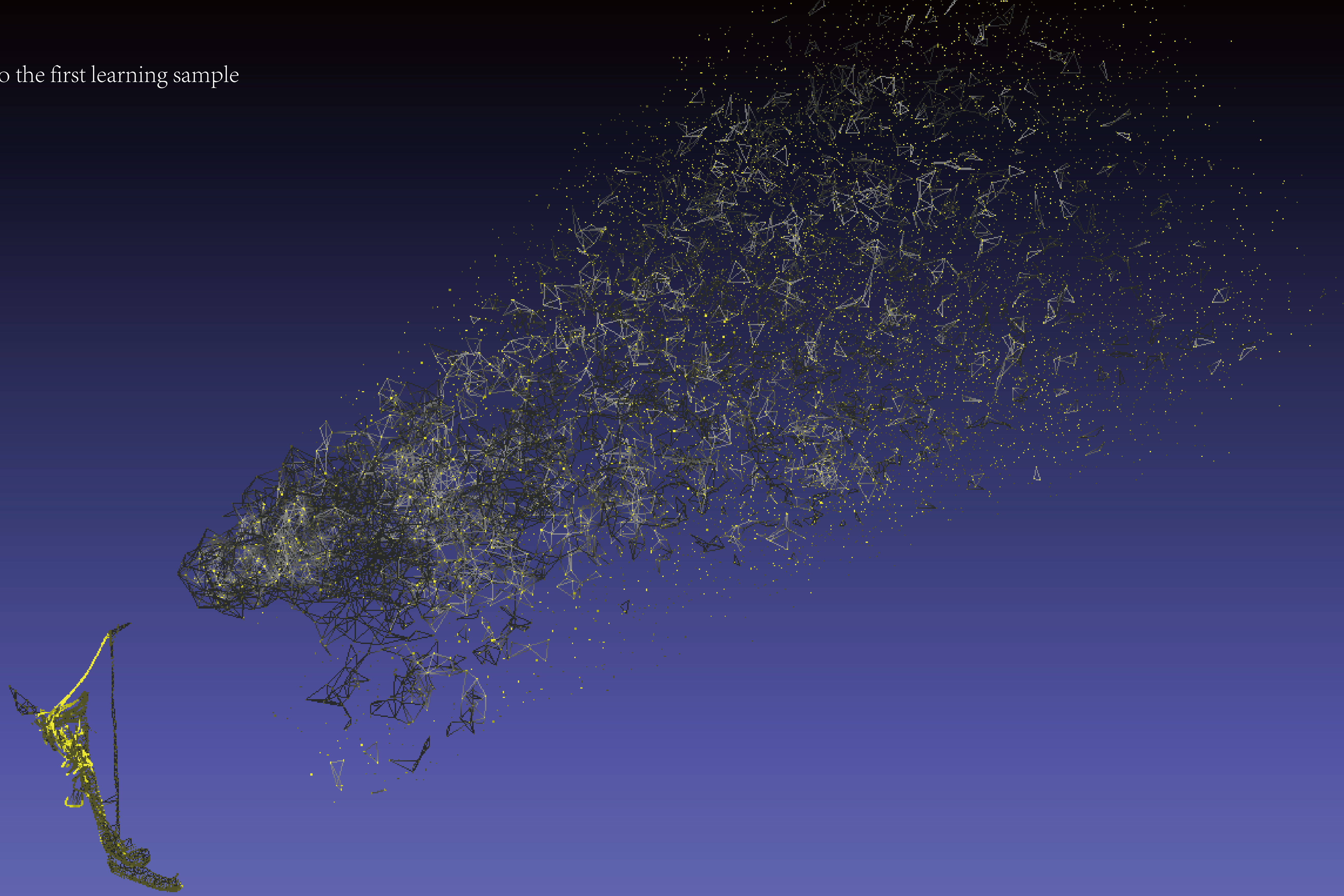
Due to the lucky mistake of two different space coordinates

The process of learning perfectly described by points moving



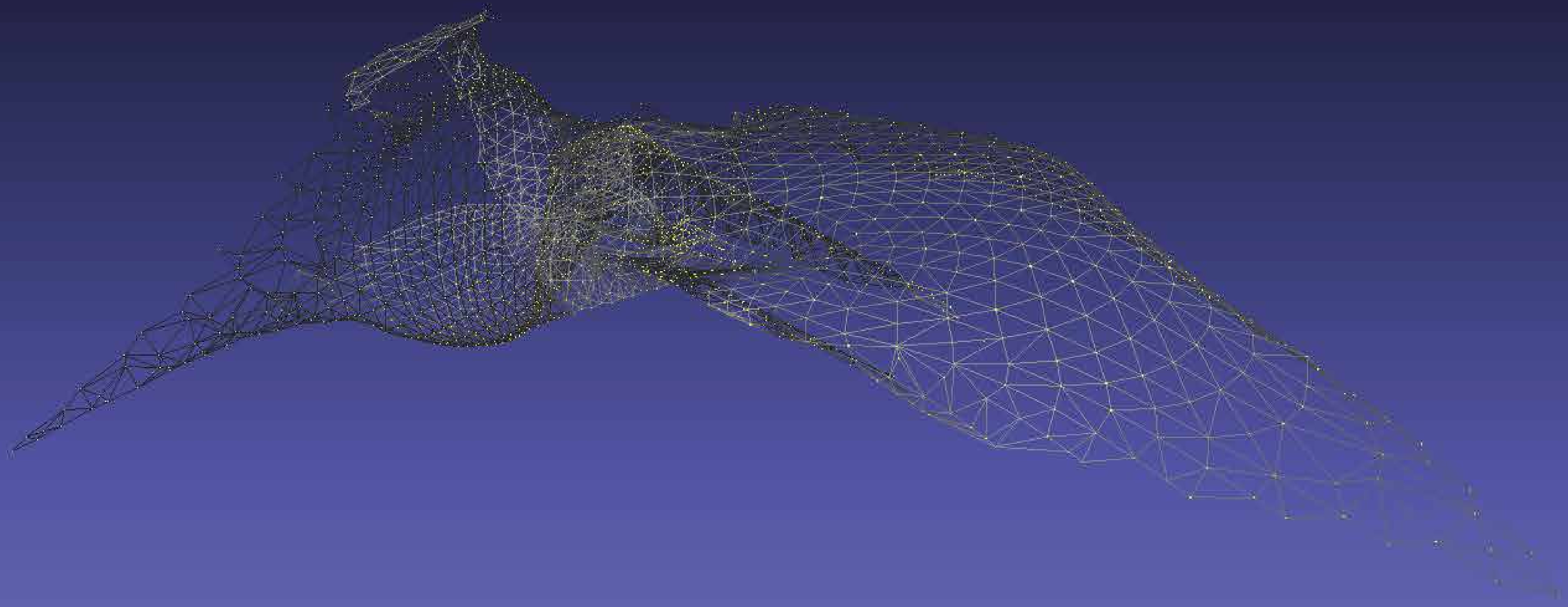
LEARNING ITERATION:1000

Lots of points are learning to the first learning sample



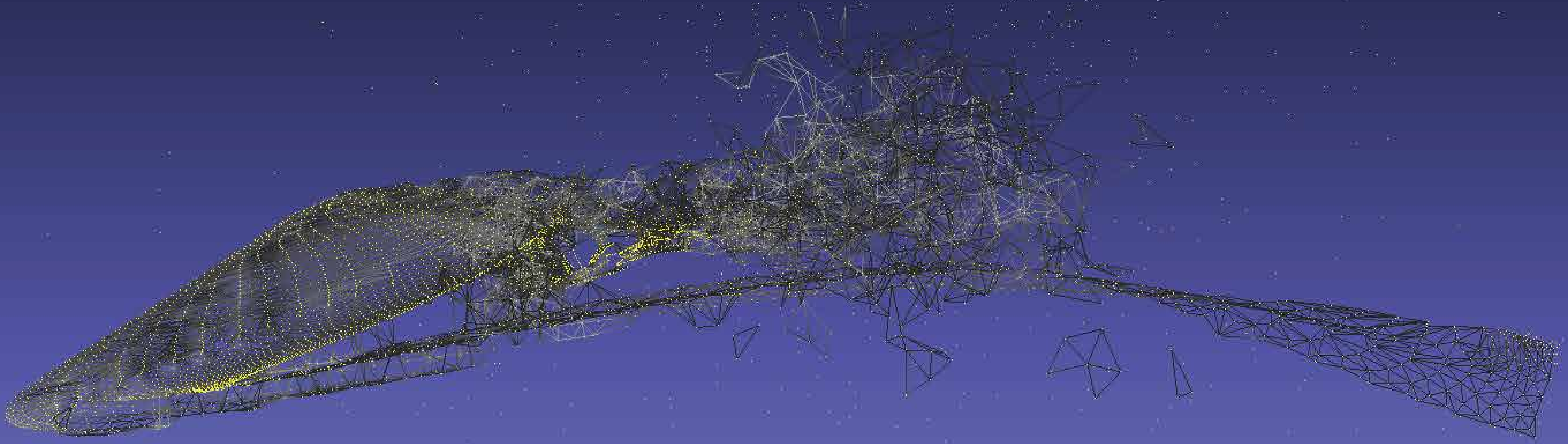
SECOND SAMPLES LEARNING ITERATION:10000

Roughly finishing the learning to first sample

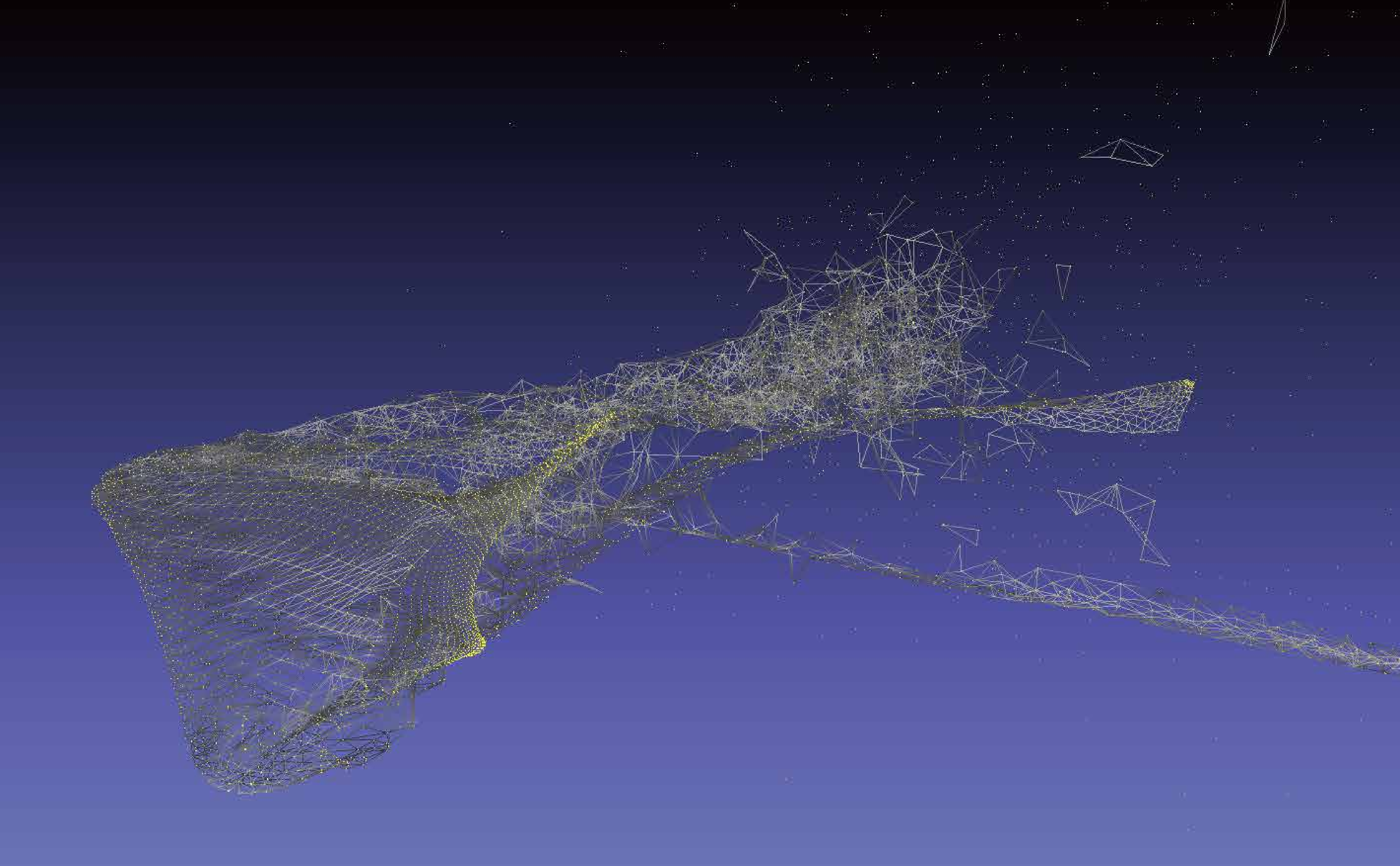


LEARNING ITERATION:200000

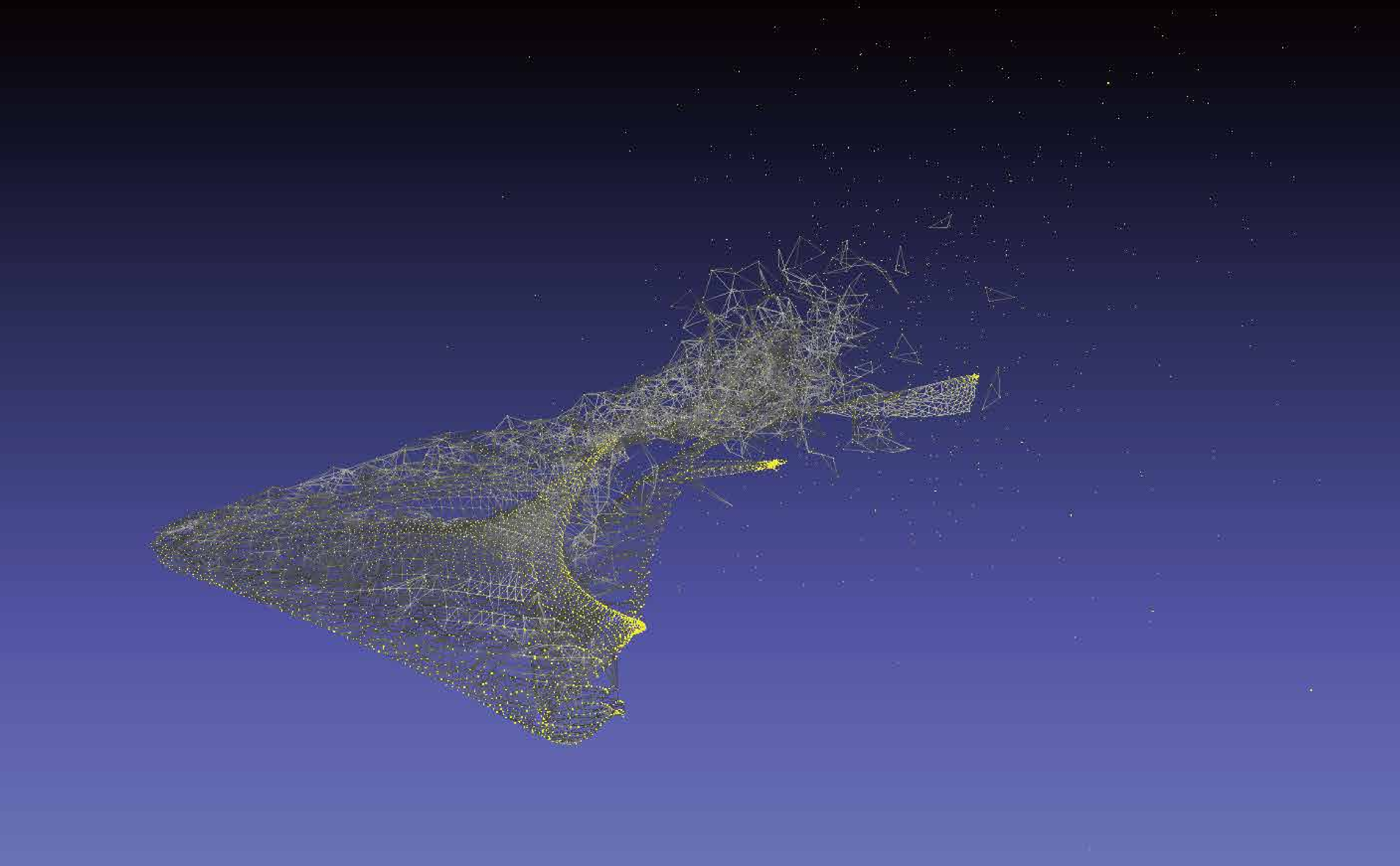
Using the result of last machine learning process to learn with second learning sample



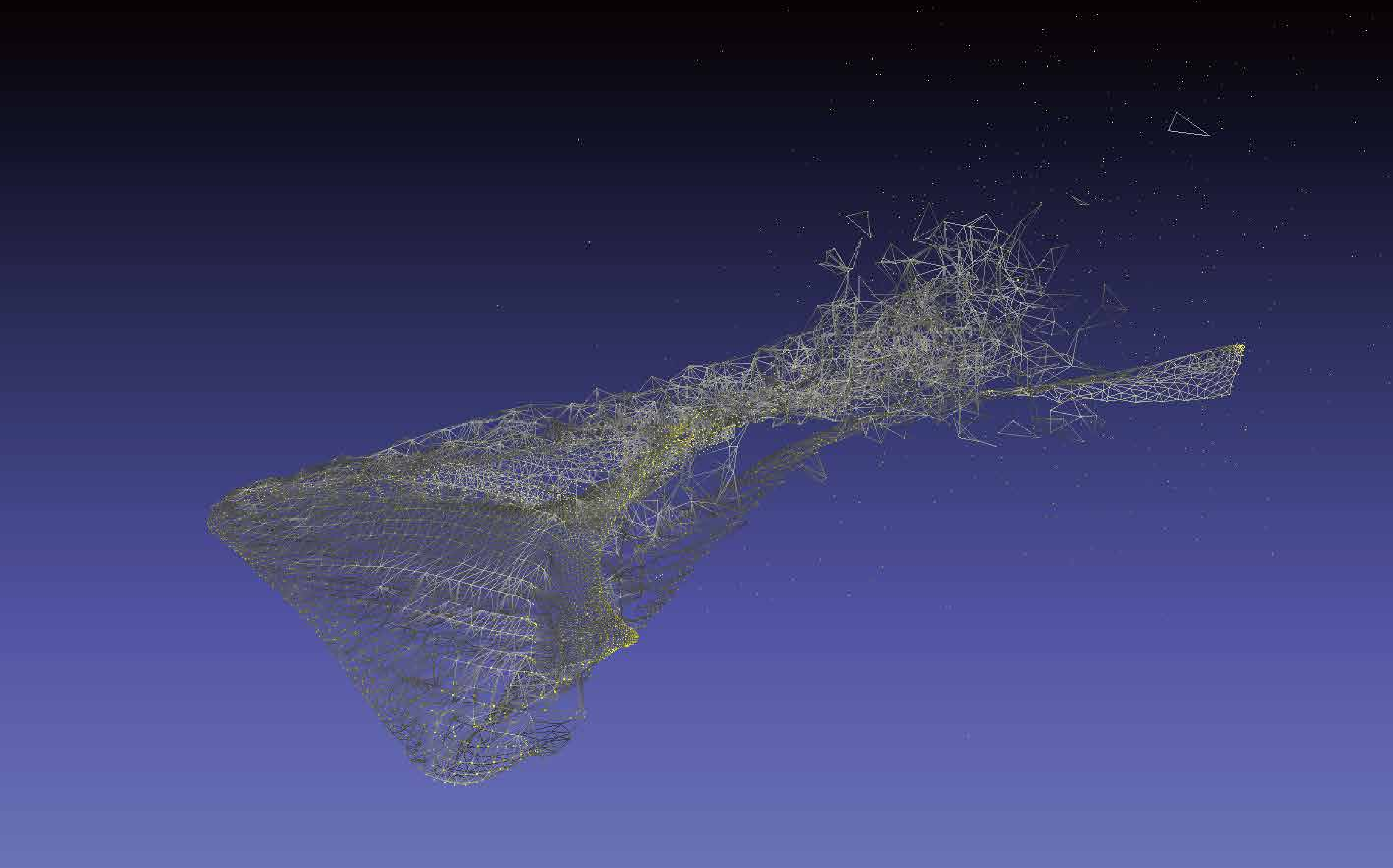
SECOND SAMPLES LEARNING ITERATION:2000



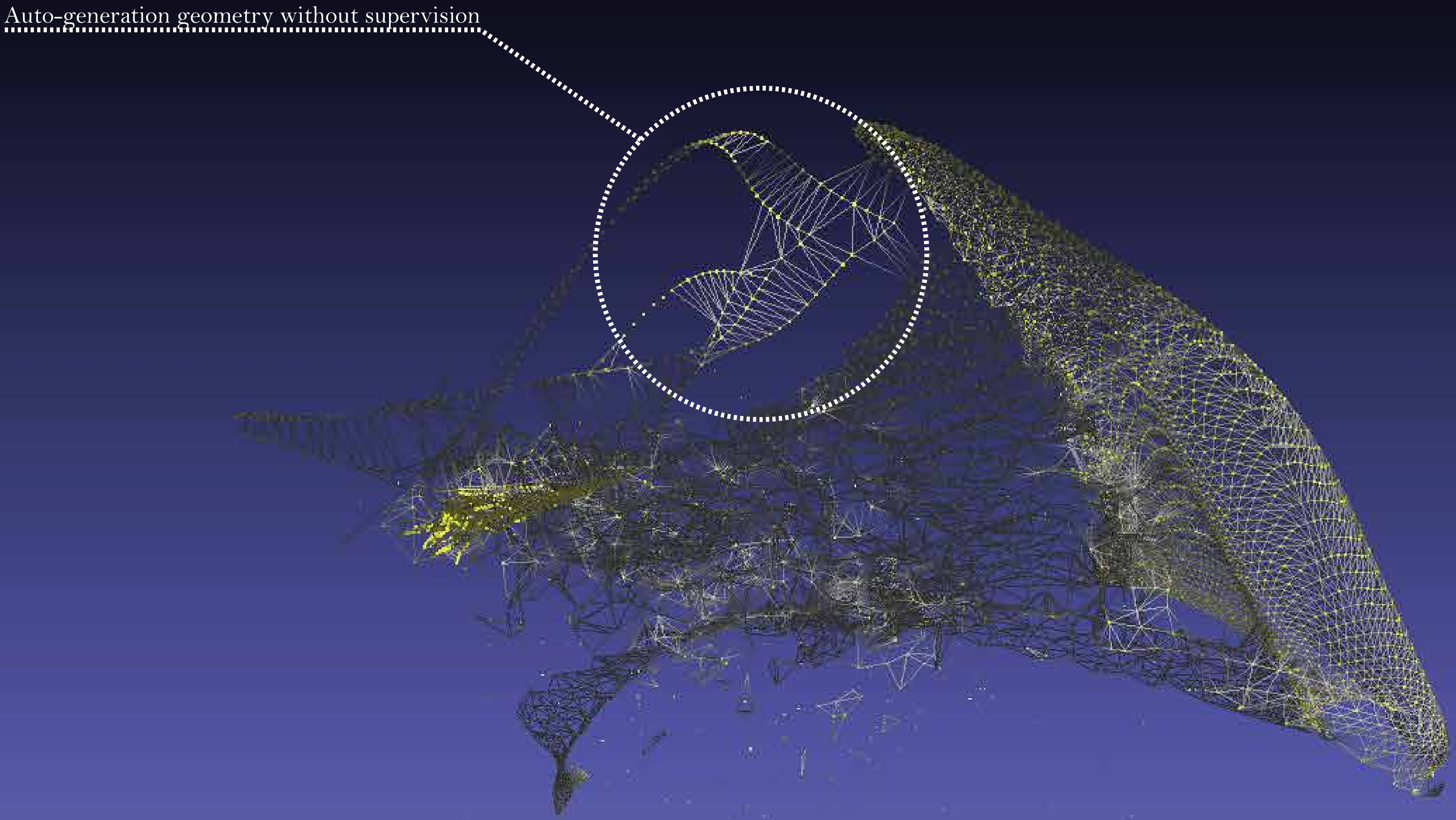
SECOND SAMPLES LEARNING ITERATION:10000



SECOND SAMPLES LEARNING ITERATION:40000



SECOND SAMPLES LEARNING ITERATION:50000



LEARNING ITERATION:200000

Form

"Where does the individuality of such a building begin and on what does it depend? Clearly it depends more on its form than on its material..."
---Aldo Rossi, The Architecture of The City, 1966

"the architectural artifact is conceived as a structure and that this structure is revealed and can be recognized in the artifact itself. As a constant, this principle, which we can call the typical element, or simple the type, is to be found in all architectural artifacts. It is also then a cultural element and as such can be investigated in different architectural artifacts..."
---Aldo Rossi, The Architecture of The City, 1966

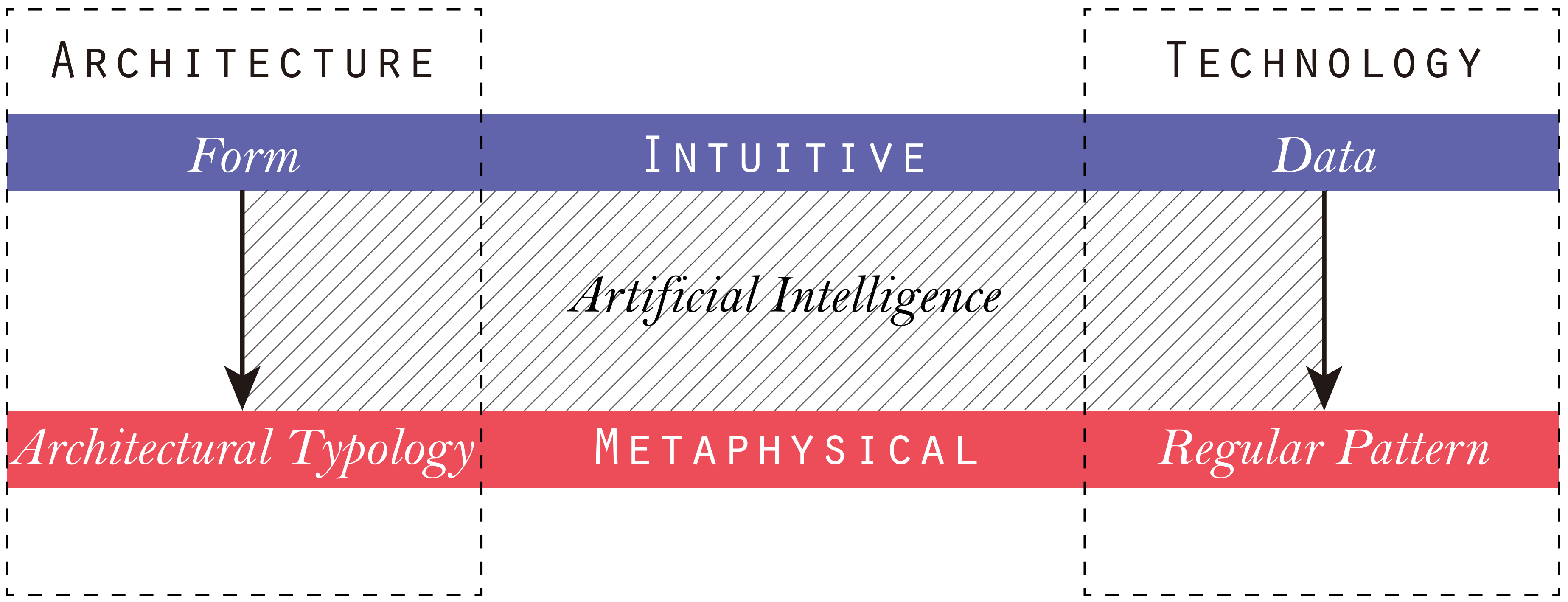
"... I would define the concept of type as something that is permanent and complex, a logical principle that is prior to form and that constitutes it."
---Aldo Rossi, The Architecture of The City, 1966

Architectural Typology

Learning instinct of human being is seeking COMMON INFORMATION

Artificial Intelligence is learning FULL-SCALE INFORMATION

Purely Architecture Value



Architecture

Technology

Form

Intuitive

Data

Artificial Intelligence

Architectural Typology

Metaphysical

Regular Pattern

Quotation

[1] Bill Bryson, *A Short History of Nearly Everything*, 2003

[2] H.P. Lovecraft, *The Call of Cthulhu*, 1928

[3] Mark Foster Gage, *Killing Simplicity: Object- Oriented Ontology in Architecture*, 2015

[4] Martin Heidegger, *Being and Time*, 1927

[5] Graham Harman, *The Quadruple Object*, 2011

[6] Mario Carpo, *Breaking the Curve: Big Data and Design*, 2017

[7] Mario Carpo, *The Alphabet and The Algorithm*, 2011

[8] Mario Carpo, *The Second Digital Turn: Design Beyond Intelligence*, 2017

[9] Jose Algeciras Rodriquez, *Trained Architectonics*, 2016

[10] Michael Hansmeyer, Benjamin Dillenburger, *Mesh Grammars: Procedural Articulation of Form*, 2013

[11] Carraher E., *Parameters of a digital design foundation*, ACADIA Regional 2011: Parametricism (2011): 99-107.

[12] Abdelmohsen, Sherif M, *Reconfiguring Architectural Space Using Generative Design and Digital Fabrication: A Project Based Course, Blucher Design Proceedings*, 1, no. 7 (2013): 391-395.

[13] Nagy, Danil, Damon Lau, John Locke, Jim Stoddart, Lorenzo Villaggi, Ray Wang, Dale Zhao, and David Benjamin, *Project Discover: An Application of Generative Design for Architectural Space Planning.*

[14] Lin, S. H., and D. J. Gerber, *Designing-in performance: Evolutionary energy performance feedback for early stage design, Proceedings of the Building Simulation* (2013).

[15] Lin, Shih-Hsin Eve, and David Jason Gerber, *Designing-in performance: towards cloud based simulation and multidisciplinary design solution space search, In Proceedings of the Symposium on Simulation for Architecture & Urban Design*, p. 1. Society for Computer Simulation International, 2013.

[16] Coates, Paul, Tom Appels, Corinna Simon, and Christian Derix, *Current work at CECA*, (2001).

[17] Dincer, Ahmet Emre, Gülen Çağdaş, and Hakan Tong, *A Digital Tool for Customized Mass Housing Design.*

[18] Broughton, Terence, A. Tan, and Paul S. Coates, *The Use of Genetic Programming In Exploring 3D Design Worlds A Report of Two Projects by Msc Students at CECA UEL*, (1997).

[19] Biao, Li, and Li Rong, *Searching generative methods based on building environments*, (2011).

[20] Caldas, Luisa G., and Leslie K. Norford, *Shape generation using pareto genetic algorithms: integrating conflicting design objectives in low-energy architecture, International journal of architectural computing*, no. 4 (2003): 503-515.

[21] Aldo Rossi, *The Architecture of The City*, 1966

[22] Patrik Schumacher, *Parametricism 2.0: Rethinking Architecture's Agenda for the 21st Century*, 2016

[23] Kostas Terzidis, *Algorithmic Architecture*, 2016

[24] Charles M. Eastman, Spatial Sythesis in Computer-Aided Building Design, 1975

[25] Antoine Picon, *Digital Culture in Architecture*, 2010

[26] William Pena, *Problem Seeking: An Architectural Programming Primer*, Third Edtition,1987

[27] John A. Bullinaria, *Self Organizing Maps: Fundamentals Introduction to Neural Networks: Lecture 16*, 2004

Code Citation

[1] Code of Self Organizing Maps, Tom Germano, March 23, 1999

[2] Kohonen, T., Self-Organization and Associative Memory, New York: Springer-Verlag, 1988

[3] Kohonen, T., Self-Organization Maps, New York: Springer-Verlag, 1997