Mesh Generation for Laser Scanned Point Clouds

Janos Mate Logo
Department of Photogrammetry and Geoinformatics
Budapest University of Technology and Economics
Budapest, Hungary
logo.janosmate@epito.bme.hu

Vivien Poto
Department of Photogrammetry and Geoinformatics
Budapest University of Technology and Economics
Budapest, Hungary
poto.vivien@epito.bme.hu

Nikol Krausz
Department of Photogrammetry and Geoinformatics
Budapest University of Technology and Economics
Budapest, Hungary
krausz.nikol@epito.bme.hu

Arpad Barsi
Department of Photogrammetry and Geoinformatics
Budapest University of Technology and Economics
Budapest, Hungary
barsi.arpad@epito.bme.hu

Abstract— **For autonomous vehicles, environmental information is extremely important. While we can generate data quickly and efficiently, but research for processing is still have a lot of opportunities to develop. Mesh generation from the point cloud is one area where development is possible.**

Keywords—mesh generation, autonomous vehicles, point clouds

I. Introduction

The mesh generation has been around for a very long time. One of the first applications comes from the second half of the 5th century when Zu Chongzhi used it to determine the value of π, which was considered the most accurate result for about 900 years. With the mesh applications, they have been used to simplify the complex (and under the circumstances the impossible) tasks. The problem solving by mesh generation is very fast and we can make a simplified model from the object or the road very fast. The different parameters and settings generate different meshes. With different mesh parameters, they are suitable for different tasks. One of the most important steps in the mesh generation step is to add parameters. Here we determine the type, shape, and quality of the resulting mesh. Subsequently, we will process, refine, and verify that the resulting mesh is of the desired quality.

The main test aspect of the created mesh is whether it is structured or unstructured. A structured mesh is that if there are two sets of lines and a line belongs to one intersect all the lines belong to the other set of lines, but each line only once. The advantage of this method is that it can write very fast and efficient solving algorithms, many methods are known for their solution and the quality of the elements can be well controlled. In the case of an unstructured or free mesh, there is no restriction on the shape of the elements and no specific pattern is used. In this case, the mesh is made up of a multitude of disorganized polygons.

The advantage of structured mesh is that connection information can be clearly deduced only from cell numbering. Any existing mesh is easy to be stored and managed on a computer. [1] However, it has the disadvantage that the orthogonality and slenderness of the cells must be within certain limits, which makes a mesh of complex geometries more difficult and less effective in locally fine meshes.

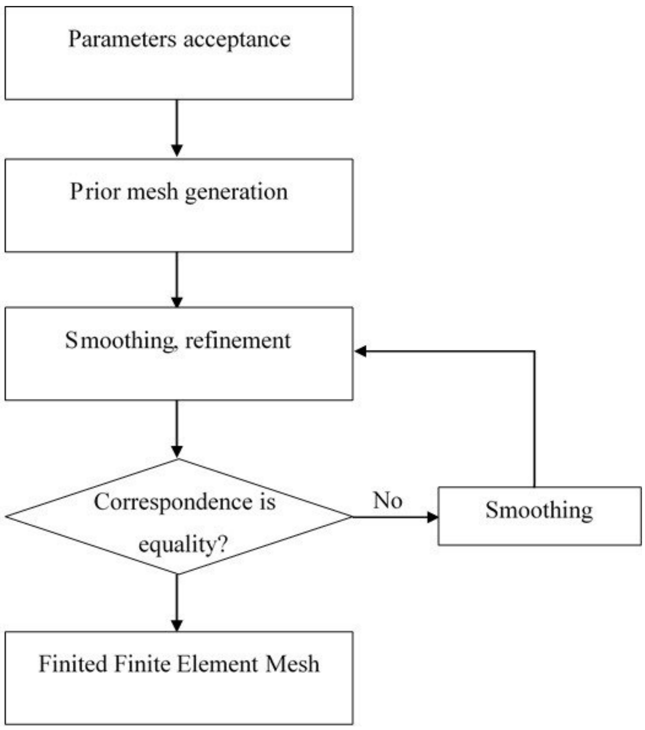

Fig. 1. Mesh generation steps

The advantages of unstructured meshing are that they are easy to network with complex geometries and are very effective for locally fine meshes. The disadvantage is that the connection information must be stored and retrieved separately for each cell, and it is more difficult to be stored and manipulated by the computer. [2]

II. Mesh types in engineering practice

We have a lot of types of mesh methods. The triangle and the quadrilateral are the most common methods, but the triangle is much more popular in computer graphics than quadrilateral. One of the best-known subdivision types is the Voronoi cell. Producing a Voronoi cell is a simple but at the same time slow process. If we connect the starting points of the adjacent convex polygons thus obtained, we get the Delaunay triangles.

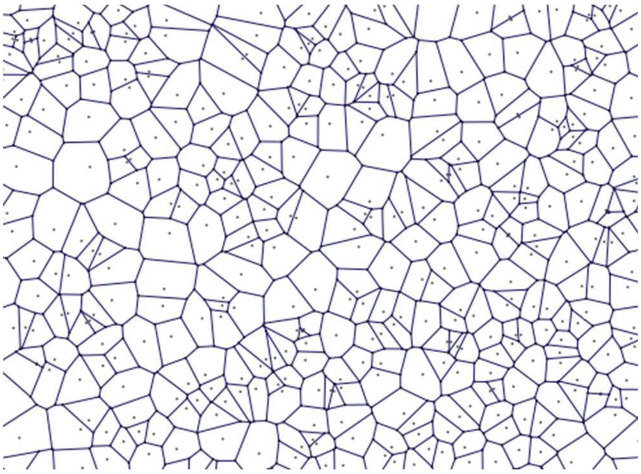

Fig. 2. Voronoi cells

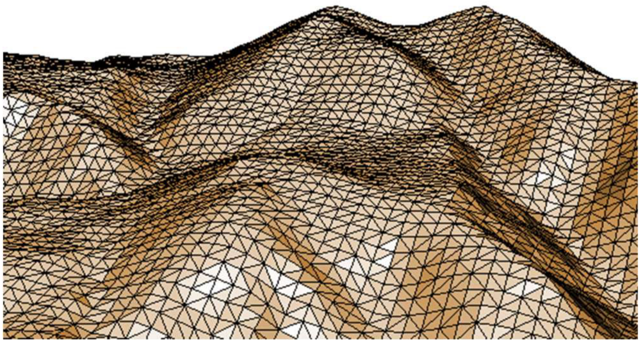

Fig. 3. Delaunay triangulation

Centroidal Voronoi tessellation is a special type of Voronoi mesh where Voronoi cells are centered on the center of gravity of each element. This road is well suited for cloud computing, with the disadvantage that it requires global parameterization and can be used on a limited surface. The other basic method of subdivision is the quadrilateral mesh. In terms of applications and speed, this method can provide a better solution than the Voronoi triangle, although the debate and results in the scientific community are fairly divided. By using this, it can more easily follow the features of the lines of the represented shape and also better adapt to deformations. [3]

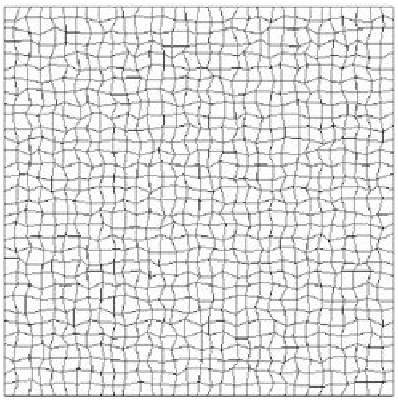

Figure 4. Quadrilateral mesh

In addition to basic meshing methods, there are many newer methods. Newer meshes include the Blossom-Quad mesh. This is a non-uniform quadrilateral mesh method, mixing triangular meshes with quadrilateral meshes, so the mesh created by this method is better for the shape of the element and also for the efficiency of the size of each field.

It needs a little longer time; first a pure triangle mesh is created, and after a quadrilateral transformation happens from the triangle when it is possible. [4]

Another solution examines the use of hexagons when processing point clouds. As a 3D scanners and imaging tools evolve at a faster rate, programs that can process them need to evolve, or a different solution must be found so that higher quality clouds with higher storage requirements and more computing can be processed with the same or even better solution than other programs. This hexagonal mesh method is used to represent surfaces constructed with flat glass or metal panels. The basis of this method is also based on triangulation, first the algorithm calculates the regular triangulation of the surface, and then converts it into a specific shaped P-Hex mesh, which fits the surface much better than a triangular mesh of the same size would have been generated.

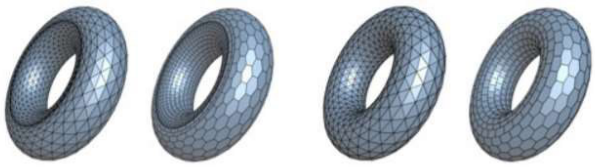

Fig. 4. P-hex

It has a lot of variation, and requires further research to find a better way to make a mesh model. [5]

III. Pilot Site for Mesh Research

The sample was made in a section of Budafoki Road in Budapest. The survey was conducted with a car, and we wanted to test what a self-driving vehicle could see if it had a suitable sensor. The original measured point cloud contained more than 6 million points, which was cleaned down to 4 million.

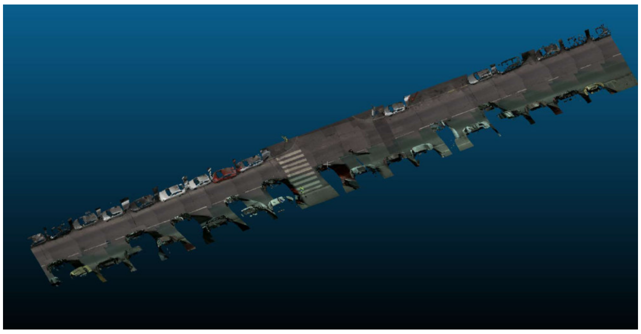

Fig. 5. Sample road

One of the major drawbacks of the remaining point cloud is that there had coverage of cars and pedestrians crossing the road. In this way, it is possible to hide details that could be important in the future.

That is why it is important to use a corresponding meshing method to add details that have not been surveyed and to link them with topological information that can be used later.

In order to find the best-fitting mesh method that produces an acceptable quality mesh and an acceptable processing speed, we examined several programs and compared the results obtained depending on their configuration.

IV. Methods and Results

First, we used the CloudCompare to create a mesh. It has a very few settings, only three types; two used the Delaunay triangles with height (2.5D), and the third one is the Poisson Recon method. The first two methods are very similar, and they don't have a big difference in the results. It is very fast, and normal quality, unfortunately it cannot be used on hilly terrain or smaller radius bend. It is perfect for flat roads. The third method is the Poisson Recon. It is applicable only to closed surfaces, but we didn't have here. The other mistake was if we don't know any details about the methods of the mesh generation, but most of them are unstructured.

The second software was MeshLab. This is an open-source software, focuses on handling and processing unstructured meshes and provides a toolbox for editing, cleaning, checking, rendering and transforming this type of mesh. The major disadvantage of the program is that it only creates a triangular mesh and can use color data, but it takes longer to process than some similar programs. If the existing point cloud is first sampled and then the Ball-Pivoting method is performed, the result will be an almost good mesh, sometimes with gaps. But the obscured objects have still not been taken in, and the running time is much, so it is not suitable.

The third application was the GMSH. In this application, the mesh always will be unstructured, even so if we made a structured method. The main option is the Delaunay method, both triangular and quadrilateral, and the Advanced Front method. It has the great advantage of being able to smooth in contrast to other programs. It is also possible to change the size of the elements in the mesh, from 10 m up to 0.01 mm. The coverings were no problem either, making the mesh easy and very fast; a mesh produced using the Advanced Front method with a 3 cm element size in less than 0.07 seconds.

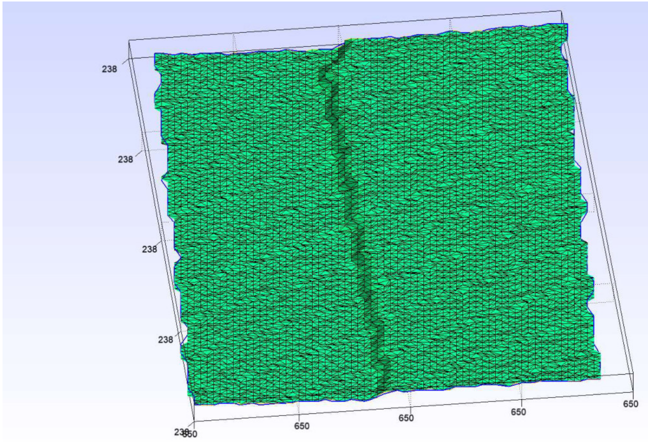

Fig. 6. Example for triangle mesh (road and sidewalk)

The deceleration time for the complete road section with the appropriate parameters is less than 2 seconds. This is a good result at such a distance.

The fourth application was the GiD. This software is capable of producing a structured and unstructured mesh. You can specify where to place the nodes in the completed mesh, generate a triangle, a rectangle, or even for a very specific examination, generate a circular mesh. The main consideration was the speed of the mesh generation it was 0.5s. Thus, the quality of the mesh and the speed of its preparation can be considered optimal.

TABLE I. Versions of the Mesh Generation

Version of the mesh methods			
Version	***Structured/Unstructured***	***Type***	***Element type***
1	Unstructured	Triangle	Linear
2	Unstructured	Triangle	Quadratic
3	Unstructured	Triangle	Quadratic9
4	Unstructured	Quadrilateral	Linear
5	Unstructured	Quadrilateral	Quadratic
6	Unstructured	Quadrilateral	Quadratic9
7	Structured	Triangle	Linear
8	Structured	Triangle	Quadratic
9	Structured	Triangle	Quadratic9
10	Structured	Quadrilateral	Linear
11	Structured	Quadrilateral	Quadratic
12	Structured	Quadrilateral	Quadratic9

Here, high processing speeds are quite significant at the expense of item size. There are several solutions available to keep the item size at the desired value. The simplest one is a better-performing computing tool. If this is not available, increase the time or reduce the size of the working area.

TABLE II. MESH GENERATION PARAMETERS (10M × 7M)

Details of the versions:				
Version	*Item size (m)*	*Node*	*Item number*	*Production time (s)*
1	0.35	78498	19472	0.5
2	0.35	109358	19502	0.5
3	0.35	109358	19502	0.5
4	0.45	85374	16612	0.5
5	0.5	119926	16524	0.5
6	0.5	136450	16524	0.5
7	0.4	78422	19320	0.5
8	0.5	108710	19178	0.5
9	0.5	108710	19178	0.5
10	0.7	85279	16517	0.5
11	1	120427	16691	0.5
12	1	137118	16691	0.5

Taking the sample into account, the time spent on the fastest triangle (version 1) and quadrilateral (version 5) for the complete road section and other data:

The quadrilateral mesh had a size of 0.2 m and contained 57 976 elements, while 688 013 nodes had a generation time of 61 seconds, the triangular mesh had a size of 0.1 m, 395 841 elements and 829 238 nodes. This generation took 200 seconds.

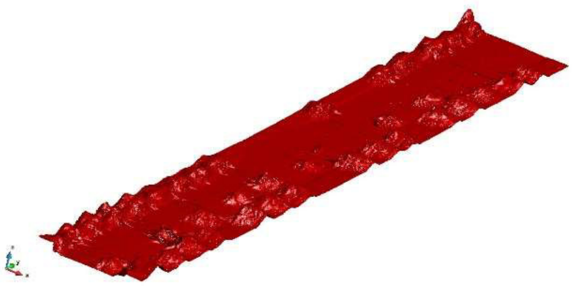

Fig. 8. Quadrilateral mesh generation

V. CONCLUSION

Each software used different methodologies, so it's hard to compare these different methods. The first and the second was used mainly for scanning a large number of point clouds, and reliably and accurately for small sections. Its disadvantage is that it uses only one method, it only works with the triangle method, and it is not clear viewing the mesh what method it came from. In the second method, the processing speed was also inadequate and it contains a lot of little holes. The third method has the advantage of adding numerous parameters, all are to be set manually. The downside is that any point cloud is impossible to use, and must be shared with another application. The fourth application has more settings than the previous ones; it works great for creating a mesh, and can be used for fast and good quality mesh. The main aspects include how we generate point clouds and define parameters in relation to the environment.

ACKNOWLEDGMENT

The research reported in this paper was supported by the Higher Education Excellence Program in the frame of Artificial Intelligence research area of Budapest University of Technology and Economics (BME FIKP-MI/FM).

REFERENCES

[1] X. Wang, X. Ying, Y.-J. Liu, S.-Q. Xin, W. Wang, X. Gu, W. Mueller-Wittig, and Y. He, "Intrinsic computation of centroidal Voronoi tessellation (CVT) on meshes," Computer-Aided Design, vol. 58, 51–61, 2015

[2] Blacker TD, Stephenson MB. Paving: A new approach to automated quadrilateral mesh generation. International Journal for Numerical Methods in Engineering 1991; 32:811–847.

[3] Blacker TD, Stephenson MB. Paving: A new approach to automated quadrilateral mesh generation. International Journal for Numerical Methods in Engineering 1991; 32:811–847.

[4] J.-F. Remacle J. Lambrechts B. Seny E. Marchandise A. Johnen C. Geuzainet: Blossom-Quad: A non-uniform quadrilateral mesh generator using a minimum-cost perfect-matching algorithm International Journal for Numerical Methods in Engineering 2012; Volume 89, Issue 9, 1102-1119

[5] Wang W, Liu Y, Yan D-M, Chan B, Ling R, Sun F. Hexagonal meshes withplanar faces. Technical report TR-2008-13 (CS series) University of HongKong.