Magnet−inspired intuitive design tool for inner structures of solid models

솔리드 모델의 내부 구조를 위한
직관적인 자석 모사 디자인 도구

2021년 2월

서울대학교 대학원

기계항공공학부

감 동 욱

솔리드 모델의 내부 구조를 위한 직관적인 자석 모사 디자인 도구

Magnet-inspired intuitive design tool
for inner structures of solid models

지도교수 이 건 우

이 논문을 공학박사 학위논문으로 제출함

2020 년 10 월

서울대학교 대학원
기계항공공학부
감 동 욱

감동욱의 공학박사 학위논문을 인준함

2020 년 12 월

위 원 장 : 안 성 훈 (인)

부위원장 : 이 건 우 (인)

위　원 : 박 희 재 (인)

위　원 : 박 용 래 (인)

위　원 : 김 영 준 (인)

Abstract

Magnet−inspired intuitive design tool for inner structures of solid models

KAM Dong Uk

Department of Mechanical Engineering

The Graduate School

Seoul National University

In this study, a new intuitive design tool that allows users to interactively design the internal structure of a given outer boundary is proposed. This study is inspired by the behavior of conductive particles under magnetic force. As the conductive particles are dense near from the magnet, the proposed approach yields denser Voronoi cells near from the magnet−like control points that are moved interactively by the designer. By describing the relation between the proposed design tool and the generated Voronoi structure, the parameters the user can interact with are also suggested. The inner structures of soild models can be created and modified by adjusting the location and number of control points and these parameters. In

addition, each Voronoi cell is modified to have a skin of desired thickness by a subdivision. Two applications of this design tool are proposed. The first proposed application is to control the center of gravity of the part, and the second is to improve the mechanical properties of certain areas of the part. In conclusion, we propose a new concept for a design tool that can lead the changing manufacturing paradigm with additive manufacturing.

Table of Contents

List of Tables

List of Figures

Chapter 1. Introduction

Recently, 3D printers have made it possible to easily create complex shapes that are difficult to make with conventional machining process. A typical 3D slicer program automatically fills the inside of a given mesh model with two options: density value and basic shape. However, these general 3D slicers provide only the regular type infill option in which the 2D shape is stretched in the stacking direction, as shown in Fig. 1. Therefore, the current slicer cannot fully utilize the advantages of the internal complex shape.
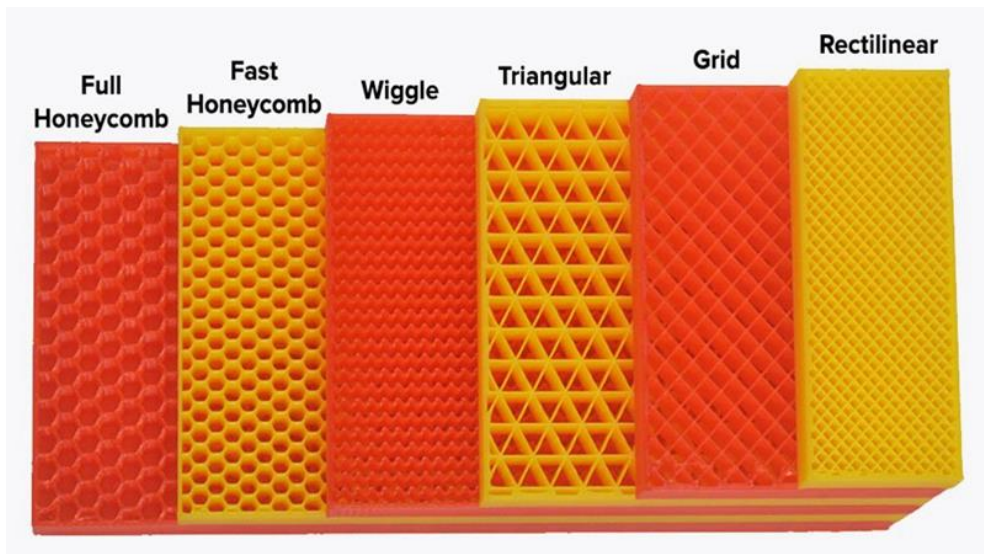
Fig. 1. Various infill structures used in typical 3D slicers

Therefore, many studies have been conducted to utilize complex internal structures. For example, studies that improve the mechanical properties of solids against weight by removing insignificant regions [1, 2], methods of making sparse structures with good ventilation [3, 4], and studies to control the elasticity of objects using specific cell structures [5, 6] have been advanced. Regli et al. [7] mentioned that a new paradigm of the material structure modeling process is emerging thanks to the advancement of additive manufacturing.

However, most of these studies are focused on the process of creating the optimal shape for a given condition rather than the modeling process in which the user can interact with and change the structure.

Complex internal structures to a certain extent can be modeled using existing computer-aided design(CAD) tools such as Solidworks or CATIA. These solid modelers have been added design functions considering the traditional machining process. Consequently, these are mainly strong in modeling the external shape of the product. However, it is extremely tedious and time-consuming work for users to deploy them to create complex internal shapes. That is, although the fabrication technology capable of creating a complex shape has been highly developed, there has not been enough research on CAD

tools that can sufficiently draw the utility of the additive manufacturing technology, owing to their limitations in modeling the internal shape. In this regard, Gao et al. [8] noted that the field of additive manufacturing cannot flourish without research on natural user interface-driven internal shape modeling methods.

Jenkins [9] pointed out that many modern engineers do not use CAD systems because of their long learning times. In particular, according to Jenkins, CAD tools that are easy to learn and use will have a major impact on productivity. For this reason, traditional CAD tools have also been changed so that non-CAD experts can learn and intuitively use them in shape design [9-11].

However, it is difficult to create various models used in additive manufacturing in current CAD systems. Azman et al. [12] evaluated current CAD tools through the design of cellular structures. In this research, it took more than 1.5 hours to design repeating patterns of cellular structures using current CAD tools. In conclusion, current CAD tools were evaluated as too difficult, time consuming, and heavy for computers to handle complex designs.

It can be said that Autodesk's Netfabb is a leading alternative CAD tool for internal shape design, as shown in Fig. 2. Netfabb's basic design principle is a type of bottom-up lattice modeling method that

first specifies the shape of a unit volume and repeats it in three axes to create the whole structure. However, this method only fills the inside of the solid body with a repetitive pattern of the same unit shape, and it is difficult for the user to make the internal shape partially different or change the local density.

Therefore, in this paper, a design tool that helps general users to design an inner structure is proposed.
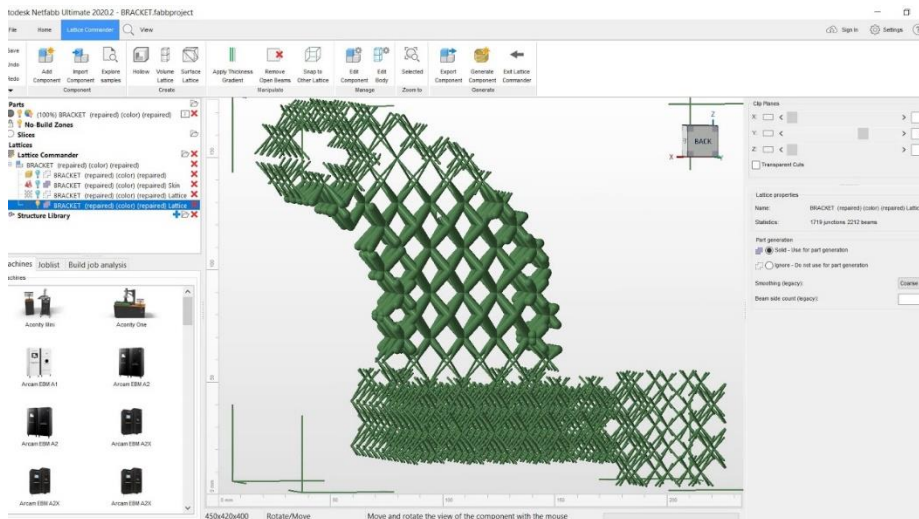
Fig. 2. Netfabb: lattice−based inner structure design tool

Chapter 2. Related Works

2.1. Related works

In this section, we review existing studies on internal design and complex shape design methods.

2.1.1. Topology optimization

Traditional topology optimization is the process of creating a shape that meets a given internal and external condition in a limited domain [13]. However, traditional topology optimization methods usually take a long time depending on the size of the domain and have the disadvantage of not being able to guarantee internal connectivity. In addition, it is difficult for the user to predict the resulting shape, and a bumpy shape may appear on the side to which an external force is applied, or an unnecessary branch shape may occur [14]. Nevertheless, topology optimization has been used in various ways to design the internal shape.

Jang et al. [15] proposed a method to mimic the human femur structure in two dimensions through topology optimization. Although these authors obtained greatly similar results to femur structure, as a limitation of this research, they mentioned that a large-scale numerical analysis method is needed.

Similarly, Jun Wu et al. [2] proposed a topology optimization method that mimics the bone structure in three dimensions. They pointed out that if there is a small difference between the fabricated product and the traditional topology optimization result, or if the external force is slightly different from the simulation input, the mechanical properties of the part may be insufficient. Therefore, to overcome these limitations, they designed a bone-like porous structure using the local density as an additional constraint.

However, although these researchers achieved speed improvement over the previous study, their method has the disadvantage of taking a considerable time to acquire the shape [16].

Larsen et al. [17] proposed a method of obtaining an optimal frame structure through topology optimization. They defined an integrated problem that considered both Michell's problem of obtaining the minimum weight of a truss that satisfies the external condition [18] and the problem of minimizing compliance property. Thereafter, they obtained a fully connected structure through an automated solution process. However, this approach is limited to 2D situations.

2.1.2. Lattice-based modeling

Daynes [19, 20] and Rahul Arora et al. [14] designed a lattice structure optimized for tensor fields obtained through finite element

method(FEM). In these studies, it is emphasized that conventional topology optimization methods require users to spend a significant amount of time on post processing, because geometry that is highly difficult to manufacture is often obtained. Therefore, the aurthor devised a method to automatically extract the Michell truss structure aligned to the stress field acquired through FEM. However, as a limitation of their method, Rahul et al. noted that there are few ways for users to control the entire structure, and further research on the user interface is required.
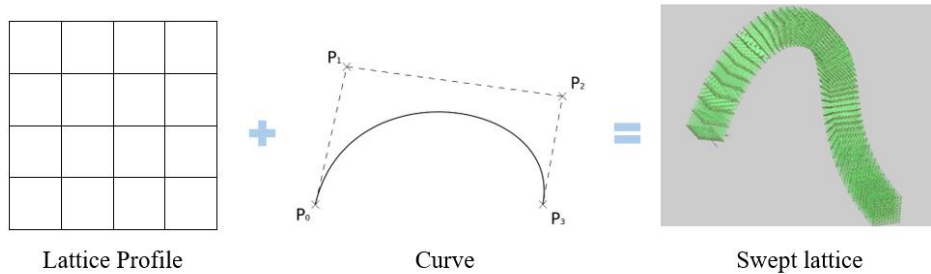
Fig. 3. Lattice sweeping method
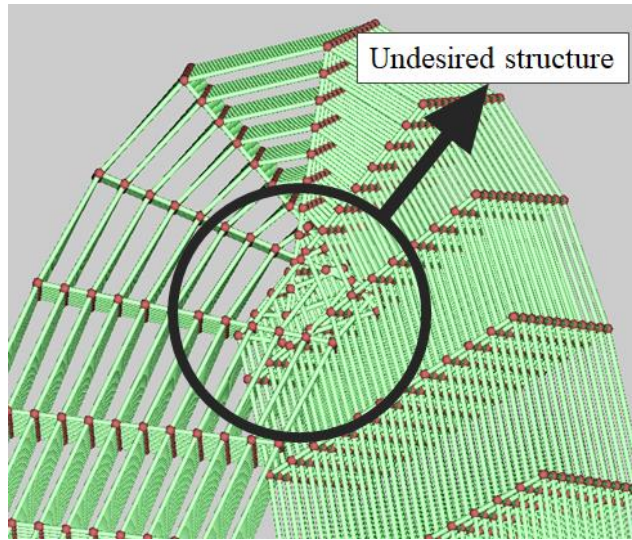
Fig. 4. Undesired swept lattice

Gupta et al. [21] studied the lattice-sweeping method that produces a 3D lattice structure with 2D grid input and a parametric path as shown in Fig. 3. Sweep operation is a commonly used design tool in conventional solid-modeling systems. It is a method of repeating an input profile along a parametric path to create a smooth 3D shape. To generate profiles repeatedly, it is necessary to calculate the local coordinates on the input parametric path at given intervals. As the method of calculating local coordinates, rotation minimizing frame [22] or steady affine motion method [23] can be applied. However, using the lattice sweep method can generate an undesired structure under certain conditions, such as a region with high curvature as shown in Fig. 4.

Bremnnan-Craddock [24] studied the process of creating a

conformal lattice structure. They firstly created a trimmed regular lattice inside the solid model. Thereafter, postprocessing was performed by wrapping a sparse structure called net−skin for the outer shell of the trimmed lattice. However, the author also pointed out that an additional process is required to change the inner density of the structure.

2.1.3. L-system based modeling

Zhou et al. [25] devised a way to design supports used in 3D printers using the L−system. L−system is a method of designing a tree−like structure by defining seed points and the growth rules of each branch. The L−system was created to mimic the growth pattern of plants [26] and has been used for digital tree generation algorithms [27, 28] and automatic city street generation processes [29]. The rules of L−system can be classified into three types: rewriting, branching and growing. However, although L−system can easily be used to create complex shapes, it has the disadvantage of it being difficult to guarantee connectivity at the end of the branches and to predict the whole generated structure.

2.1.4. Honeycomb-like 3D structure

Lin Lu et al. [30] proposed the algorithm to design the inner shape

of the honeycomb−like structure conforming to the corresponding stress field. They first calculated the stress field inside the solid model for a given external force condition. Then, the corresponding Voronoi structure was created using this stress field, and the design process was performed to create a volumetric porous structure by calculating the harmonic field between the outer surface and the center of each Voronoi cell. However, there is a limitation that the relationship between the design structure and actual mechanical properties is not strictly defined. Another limitation is that it takes an hour or more to attain the final shape.

2.1.5. Cellular structure design

Research on the cell structure design can be classified into studies on the shape design of the unit cell itself and studies on the overall shape design considering the connectivity of cells.

Aremu et al. [31] proposed a method to create a complete grid design by discretizing various cell shapes in voxel units and combining the repeated cell structure and the net−skin covering the outside. Fryazinov et al. [32] proposed a method of creating a repetitive pattern of cellular structures that was previously difficult to make through function representation.

Shanqing Xu et al. [33] studied a method of designing a cellular

structure using anisotropic values as user input. This cellular structure is basically lattice based, and authors discussed how the topology of the unit cell structure and the thickness of the lattice change the mechanical properties of designed structure. This study suggests that the shape of the internal structure is closely related with the mechanical characteristics, but the limitation as a design tool is obvious in that it is the study of a single cell structure.

Berger et al. [34] simulated mechanical properties about unit cell structures classified into six types. They concluded that a closed form structure in which cubic and octet shapes are combined can maximize the stiffness. They also stated that the use of complex cellular structures would become increasingly widespread in engineering systems.

2.2. Requirements for inner structure modeling tool

The requirements of the inner design tools of solid models presented in this study are as follows:

First, the design tool should be able to change the inner density of the solid model. In the studies related to topology optimization, complex shapes were designed using the local volume density as the constraint. The reason for using this constraint is based on the relationship between the porosity and the mechanical properties.

Zhang et al. [35] modeled the implants used for the load-bearing bone tissue reconstructions as the porous cellular structures, and then analyzed the mechanical properties of each model depending on the porosity. As a result of the analysis, it was confirmed that as the porosity increased, the compressive strength and elastic modulus decreased. Wang's result [36] also shows that the relative yield strength and modulus of elasticity decreased as the porosity of the cellular structure increased.

In addition, the study of shape design using input mechanical properties supports why users need to control the internal density of solid models.

From the modeling results through homogenization [37] or solid isotropic material with penalization [38, 39], it can be proved that there is a positive proportional relationship between mechanical

properties and internal structure density.

As an example of the actual design process, let us assume that the designer performs the internal design of T-shaped parts as shown in Fig. 5. In general, the designer expects the mechanical properties of the intersection to be important and will perform an inner structure design process that increases the density at the intersection.

Second, the design tool should be able to interact with users in real time. Each of the studies mentioned in section 2.1 achieved good results, but most of them were focused on the optimization process not the computational cost. Therefore, users have to wait a long time, ranging from tens of minutes to several hours, before obtaining the designed result.
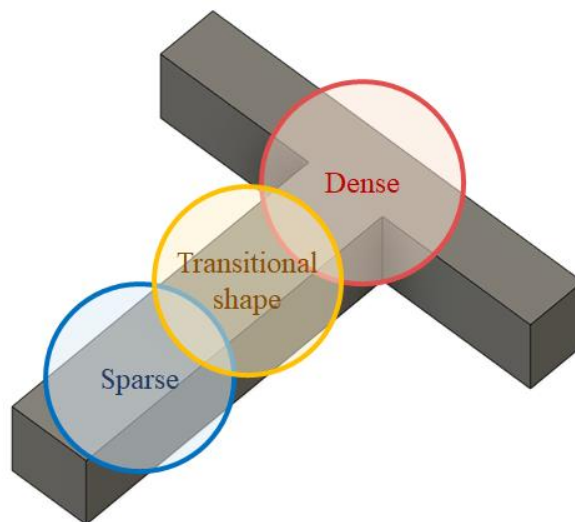
Fig. 5. T-shaped part

In terms of human-computer interaction, Goodman's study [40] shows that the shorter system response time of the graphic user interface(GUI) increases the productivity of the design process. Finally, design tools should be easy to use even for nonprofessionals. As mentioned in the introduction, even if the content of the software is well made, the impact can be reduced if the user cannot easily use it.

For example, conventional surface-modeling systems [41, 42] use control points to create and modify surfaces as shown in Fig. 6. These control points act as handles that utilize interactive graphics capabilities in modern CAD systems. Similarly, the inner design tools of solid models should also provide handles to allow users to interact with them.
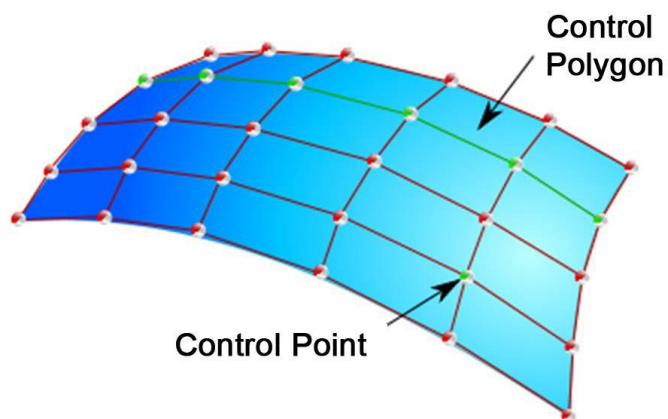
Fig. 6. Control points for Bezier/B-spline surface[41]

Chapter 3. Method

3.1. Concepts of modeling tools

Fig. 7. Phyllotaxis, visualization of magnetic field lines,

bubble structure

To create a user−friendly interior shape design tool, we took
inspiration from various shapes found in nature, as shown in Fig. 7.
These shapes are a kind of cellular structure that gradually changes
its density.
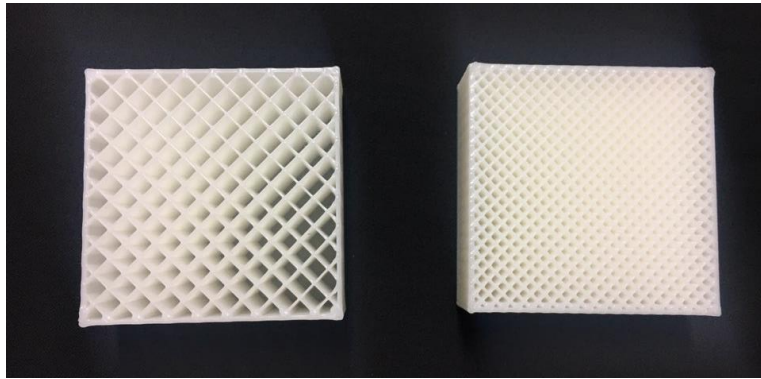
Fig. 8. Generated infill structures using a typical 3D slicer program
with various density options, density 20% (left) and 40% (right)

It can be found how the term "denser infill" is accepted in common
sense through the infill structures from a typical 3D slicer program.
Fig. 8 shows the generated infill structures using a typical 3D slicer
program with various density options. In this case, a high density of
cellular structures means that there are small but large numbers of
cells.

Initially, we assumed that each cell is a conductor particle inside the
solid model. Then naturally, the magnet-like attractors presented
themselves as a candidate of design tool for inner cellular structures.
In this case, a denser structure near attractors will be expected.

To apply this idea, it is necessary to specify the behavior between
the attractors and the conductor particles. There have been studies
[43] to use magnets to control conductor particles in practical
experiments, as well as studies [44-46] to determine the behavior

of conductor particles under the magnetic force through simulation. However, the purpose of these studies is the fabrication of the products [43, 44] or to explain the behavior of magnetic particles [46].

In particular, most simulation studies [45, 47] assume a dynamic situation and the position of the particle changes for every time step. In this case, all relations between each particle have to be defined and calculated, which involves prohibitive computational cost. Thus, a different approach is necessary to apply this idea to real-time interactive applications.

In this paper, rather than directly calculating the relation between attractors and each cell, an indirect method is used in which cellular structure divides and occupies the field generated by attractors. That is, the segmentation algorithm has been applied.

First, the stippling algorithm [48, 49], which is often used in 2D applications, is adapted to determine the location of each cell. In 2D stippling, the number of points inside of the unit area varies according to the intensity value of the input image, as shown in Fig. 9.
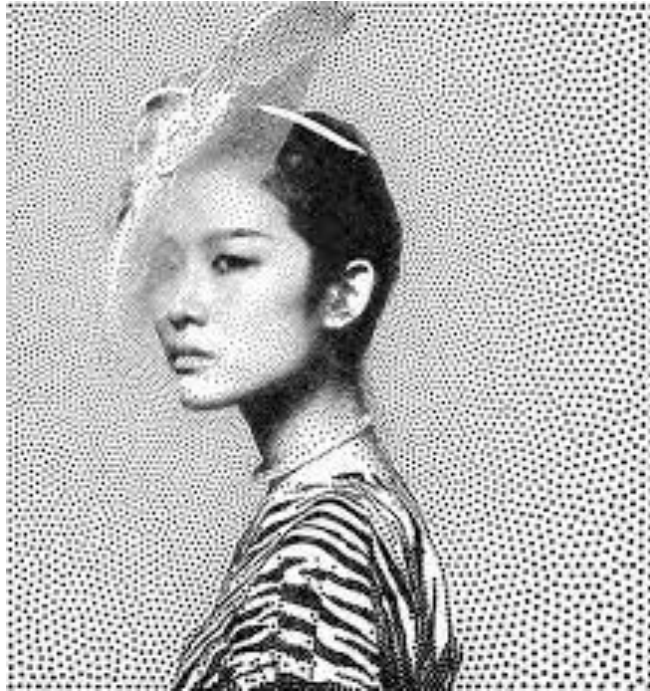
Fig. 9. An example of 2D digital stippling [49]

Points acquired through the stippling algorithm have only location information. Thus, to create a porous structure, it is necessary to generate local volume information using those points. To do this, the Voronoi tessellation algorithm [50] is utilized, as shown in Fig. 10. The Voronoi tessellation algorithm is a method of segmenting a space using bisectors between adjacent points. In this study, Rycroft's Voro++ library [51] is used.

18
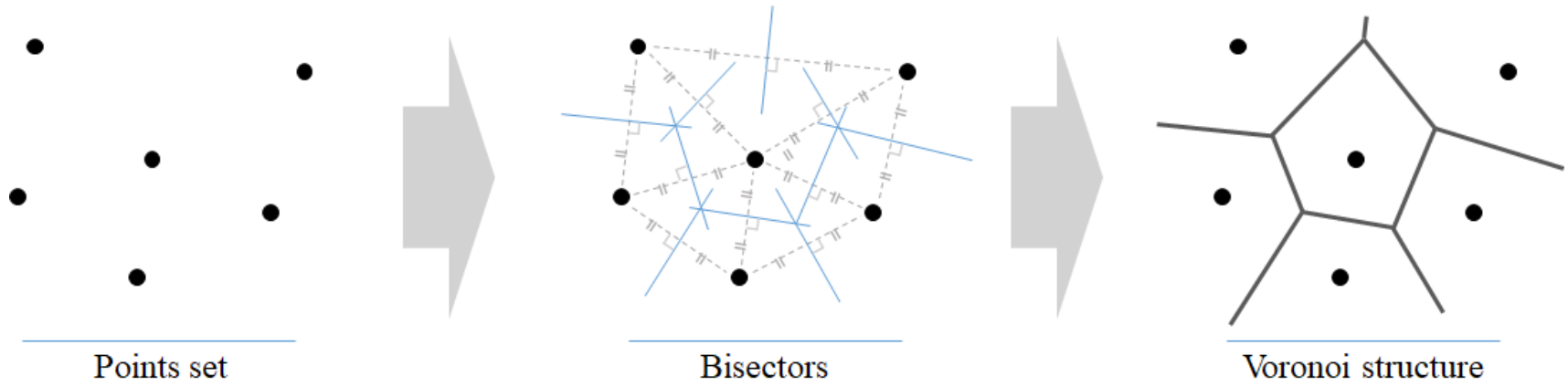
Fig. 10. The Voronoi tessellation procedure

19

The overall process of this paper is shown in Fig. 11. The first step is to load the solid model as shown in Fig. 11 (a). The user adds attractors that can control the inner field of the solid model. Depending on the added attractors, an internal field is created and modified as shown in Fig. 11 (b). After that, the user determines the number of cells inside the solid model and generates Voronoi cells according to the inner field. After that, as shown in Fig. 11 (c), each cell is scaled around its center, in other words, each cell boundary is thickened to create a fabricable porous structure. As shown in Fig. 11 (e), the user can perform the feedback design by adjusting the control points or the parameters. Fig. 11 (d) shows a fabricated inner structure through these processes.

(a) Input surface mesh

(b) Controlled density field

(c) Generated inner structure

(d) Fabricated model

Magnet like design interfaces

(e) Feedback control

Interface control

Sparser — Density control — Denser
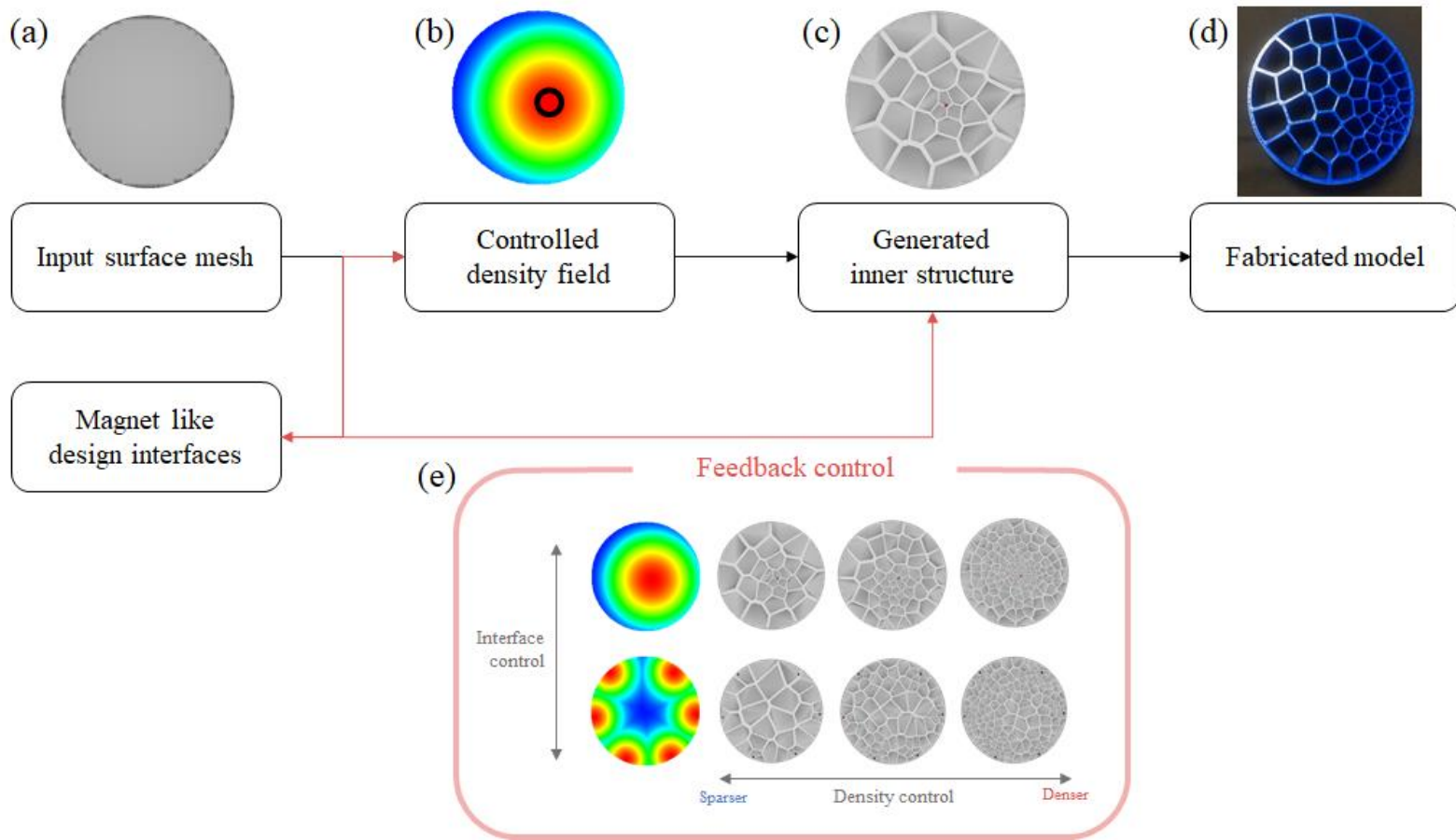
Fig. 11. Schematic diagram of this research

3.2. 3D stippling through Linde-Buzo-Gray (LBG) algorithm

Digital stippling is a method of determining the location of each stipple by evenly dividing the total value of the input scalar field. In general, 2D stippling takes a grayscale picture as its input. To apply this method to 3D application, the field values must be determined according to the attractors. In addition, the values should be able to be integrated numerically. Those methods will be described in section 3.3 and 3.4. In this section, the adapted stippling algorithm is described.

As a stippling algorithm, Lloyd's algorithm [52], which changes and optimizes the position of a fixed number of points can be utilized. However, using Lloyd's algorithm, the results may vary depending on the initial positions of the points. Therefore, in this study, the Linde−Buzo−Gray (LBG) algorithm [49], a generalized method of Lloyd's algorithm, is used. The pseudo code is Algorithm. 1.

3D stippling optimization based on LBG algorithm

Input: The number of stipple n, density field ρ, seed points $\mathbf{P_0}$

Result: The stipple positions P

Initialize;

iter = 0;

stopCondition = false;

$\mathbf{P} = \mathbf{P_0}$;

while stopCondition == false **do**

   build Voronoi diagram $\mathbf{V}$ using $\mathbf{P}$

  for each Voronoi cell $\mathbf{V_i}$ **do**

     calculate cell value $\boldsymbol{val_i} = \int \boldsymbol{\rho} \in \mathbf{V_i}$

  end

   stopCondition = true;

for each Voronoi cell $\mathbf{V_i}$ **do**

   if $\boldsymbol{val_i} < \boldsymbol{Threshold_L}$

    delete $\mathbf{V_i}$

    stopCondition = false;

else if $\boldsymbol{val_i} > \boldsymbol{Threshold_H}$

  split $\mathbf{V_i}$

stopCondition = false;

else

move $\mathbf{V_i}$

end

end

update $\mathbf{P}$ using $\mathbf{V}$

iter = iter + 1;

end

Algorithm. 1. Pseudo code for 3D stippling based on LBG algorithm

The algorithm starts with calculating the total summation of the field in the entire volume domain. Then the objective value of each cell is calculated through dividing the total value by the target number of cells. Thereafter, all the field values in each cell are integrated. A different action is operated to each cell depending on the integrated value. Fig. 12 describes the relation between each cell value and actions. If the value of the cell is greater than the objective value, the cell is divided into two cells. Otherwise, if it is smaller than the target value, the cell is removed. Otherwise, only the cell's position is adjusted. In summary, the number and location of cells are simultaneously adjusted through the LBG algorithm.

Owing to numerical errors, it is difficult to accurately match the integrated value of each cell with the objective value. Therefore, in this paper, it is assumed that the integrated value matches the objective value if it is within the range of $0.8-1.2$ times of the objective value.
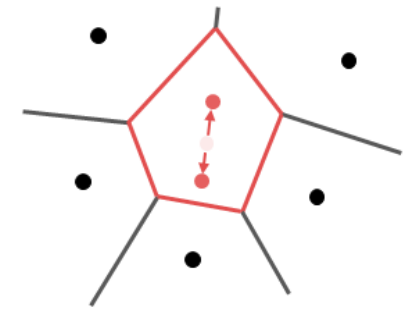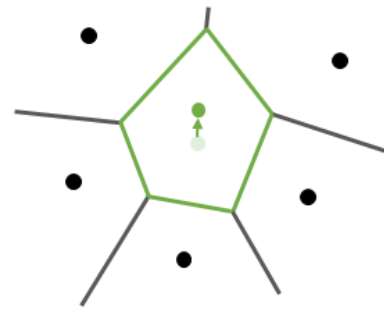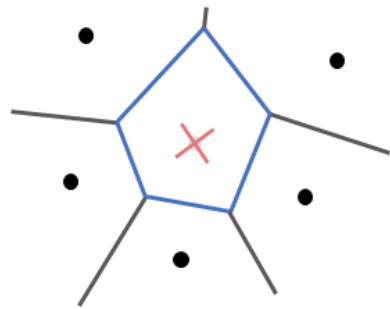
Cell value	Too low	Appropriate	Too high
Behavior	Delete	Move	Split

Fig. 12. Cell behaviors while LBG optimization

3.3. On the scalar field equation

In the stippling algorithm, the Voronoi cell and the field values are highly related. That is, in this paper, the field equation generated by attractors directly determines the size and position of the Voronoi cell. Therefore, the field equation suitable for the intuition of this research must be determined.

Given a control point, it is expected that a transitional inner structure will be designed according to the distance from a control point. Fig. 13 shows the 2D Voronoi tessellation results according to various field equations using the distance from the control point as a parameter. Generally, it can be said that in Fig. 13 (e), (f), (h), (i), the size of the cell over distance changed too quickly, and in (b), it hardly changed.
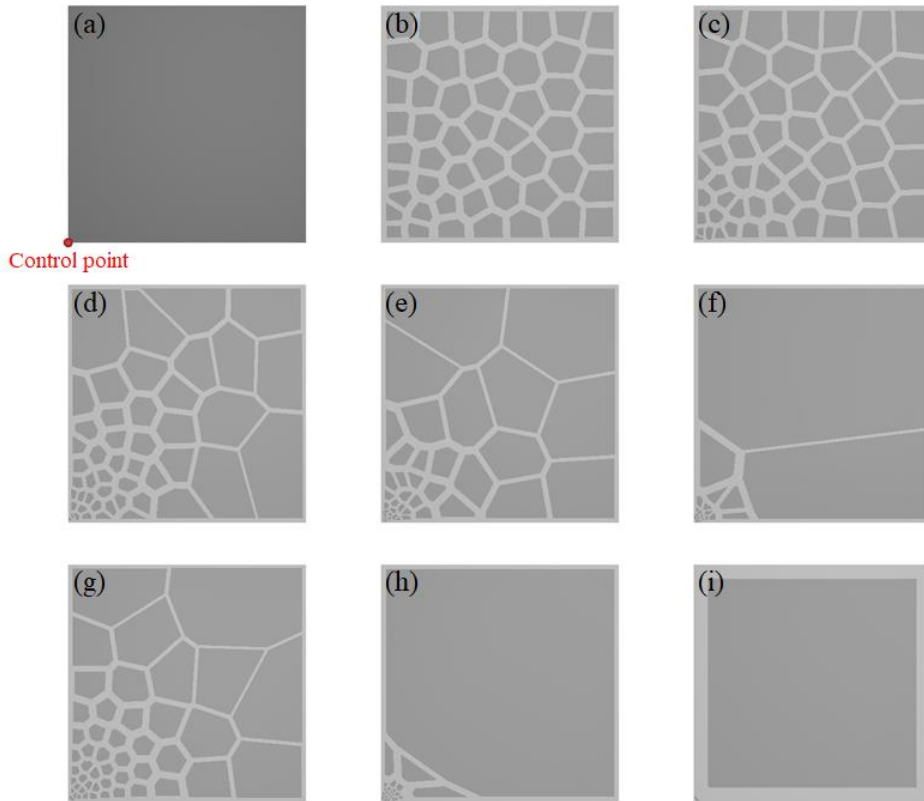
Fig. 13. 2D Voronoi tessellation results according to various field equations $\rho(r)$, (a) Experimental environment, (b) $1/r^{0.5}$, (c) $1/r^{1.0}$, (d) $1/r^{1.5}$, (e) $1/r^{2.0}$, (f) $1/r^{3.0}$, (g) $e^{-r^{0.5}}$, (h) $e^{-r^{1.0}}$, (i) $e^{-r^{2.0}}$

To assist an intuitive understanding of the relationship among control points, cells, and field equations, Fig. 14 shows a balloon that changes size according to the atmospheric pressure. As the altitude increases, the atmospheric pressure decreases. In addition, as the atmospheric pressure decreases, the balloon size increases. Consequently, a high altitude makes a large balloon size. In this way, the distance from the attractor can be a parameter of the field equation, and the cell's position and size will be determined from the field values.

Further distance   (Distance   : ↑)
Lower pressure   (Field value: ↓)
Bigger balloon   (Cell area   : ↑)

ground

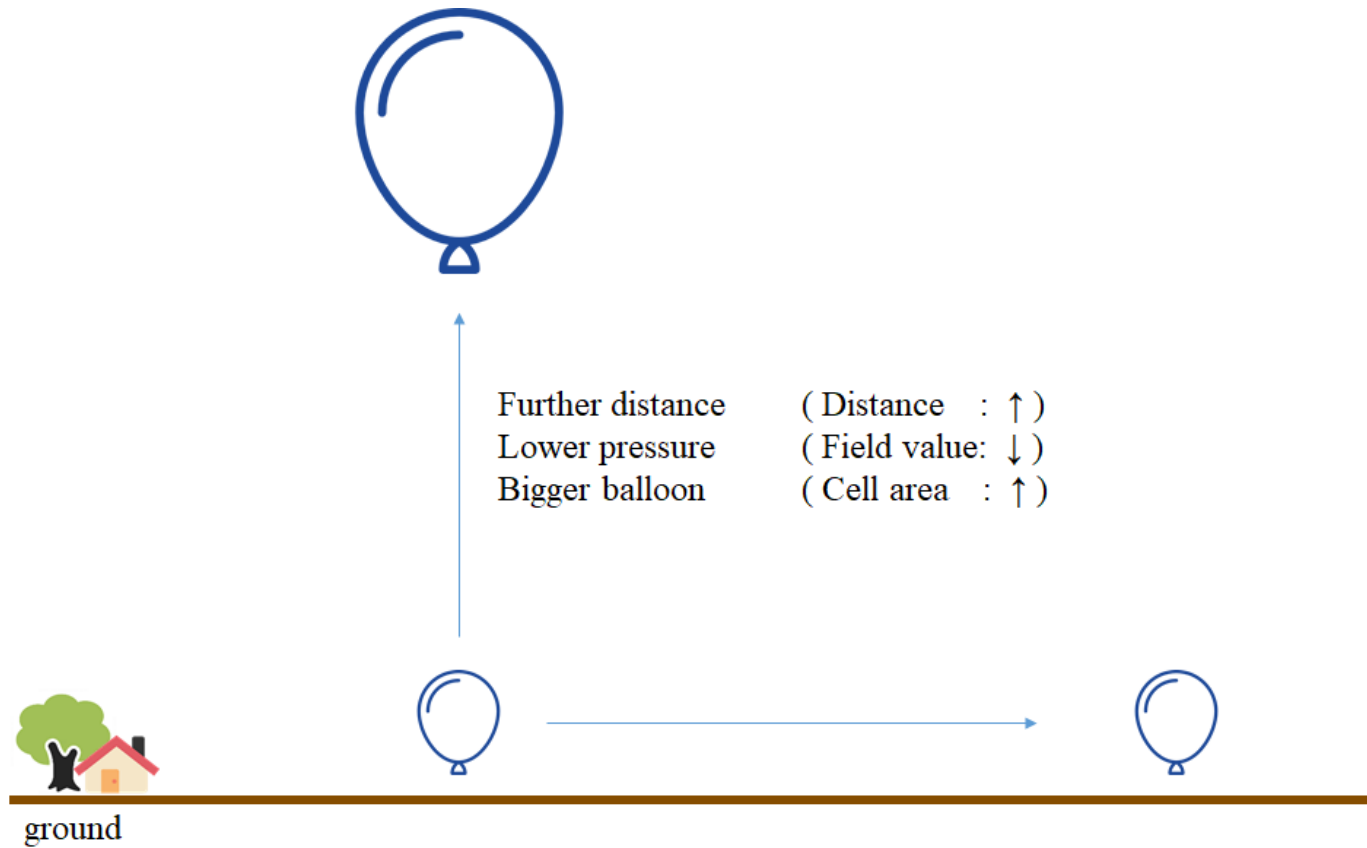Fig. 14. Balloon size change according to atmospheric pressure

3.3.1. One-dimensional case

When there is no effect of the attractor, as shown in Fig. 15 (a), it is reasonable that the subset space has the same length. On the other hand, if the length of the subset space increases linearly as shown in Fig. 15 (b), the corresponding field equation can be the criterion of the attractor's strength. The following is the process of deriving the field equation when the subset length increases linearly.

The objective function of the stippling method is to set the area so that the integration of the field values of each area is the same. That is, for the 1D case, when the field value of a specific position x is $\rho(x)$, Equation (3.1) should be satisfied.

$$\int_{x_0}^{x_1} \rho(x)dx = \int_{x_1}^{x_2} \rho(x)dx = \cdots = \int_{x_{n-1}}^{x_n} \rho(x)dx \qquad (3.1)$$

If $\rho(x)$ is a constant value ρ_{const}, Equation (3.1) can be rewritten as Equations (3.2) and (3.3).

$$\rho_{const} \times \int_{x_0}^{x_1} dx = \rho_{const} \times \int_{x_1}^{x_2} dx = \cdots = \rho_{const} \times \int_{x_{n-1}}^{x_n} dx \qquad (3.2)$$
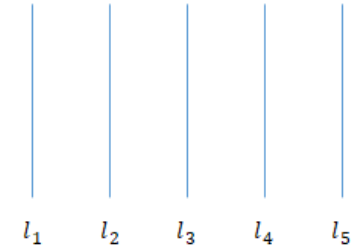
$$\rho_{const} \times l_1 = \rho_{const} \times l_2 = \cdots = \rho_{const} \times l_n \qquad (3.3)$$

Therefore, the lengths of all subset regions have the same value as shown in Equation (3.4).

$$l_1 = l_2 = \cdots = l_n \qquad (3.4)$$

In this case, the relation between the length and the partial integration of each subset is shown in Fig. 16.

(a) Evenly Divided

L

x_0 x_1 x_2 x_3 x_4 x_5

Inner space

$l_1 = \dfrac{L}{5}$ $l_2 = \dfrac{L}{5}$ $l_3 = \dfrac{L}{5}$ $l_4 = \dfrac{L}{5}$ $l_5 = \dfrac{L}{5}$

l_1 l_2 l_3 l_4 l_5

Scalar field

x

(b) Linearly Increased

L

x_0 x_1 x_2 x_3 x_4 x_5

Inner space

l_1 l_2 l_3 l_4 l_5

l_1 l_2 l_3 l_4 l_5

Scalar field

x
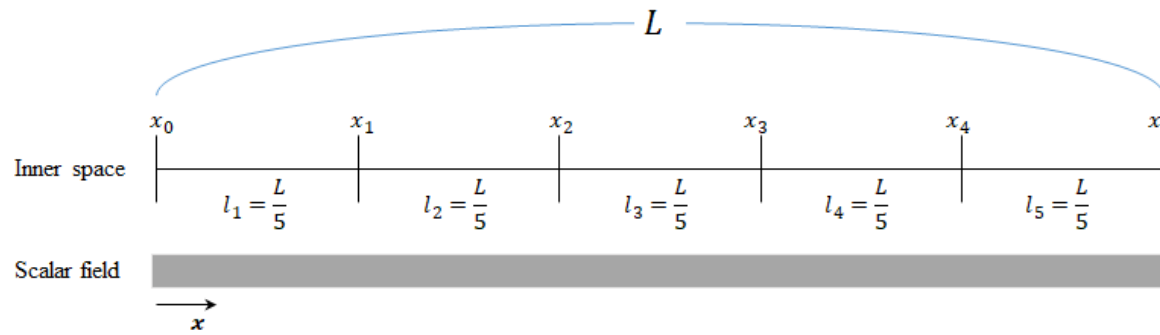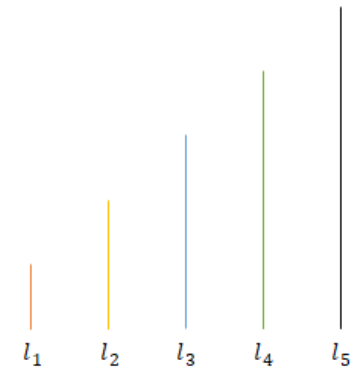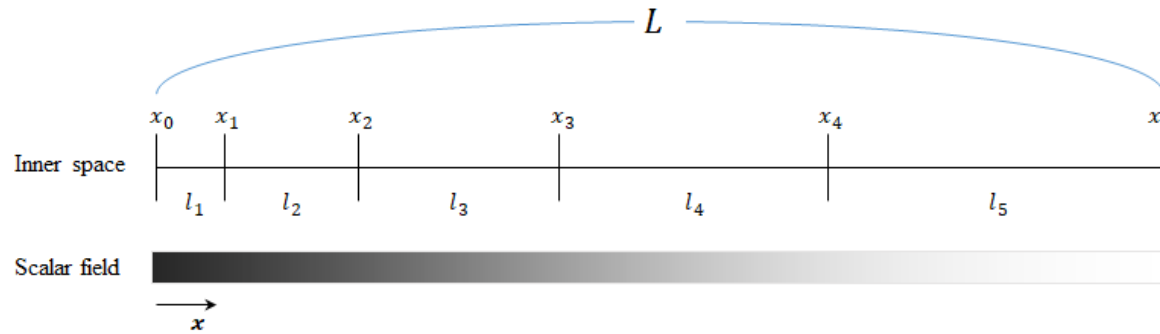
Fig. 15. 1D case example, (a) Evenly divided case, (b) Linearly increased case

$$\text{Mass of the i the segment } = \frac{Total\ Mass}{Number\ of\ particles} = \int_{x_i}^{x_{i+1}} \rho(x)dx$$

$$Length\ of\ the\ i\ th\ segment\ =\ Unit\ Length\ \times\ (i+1)\ =\ x_{i+1} - x_i$$
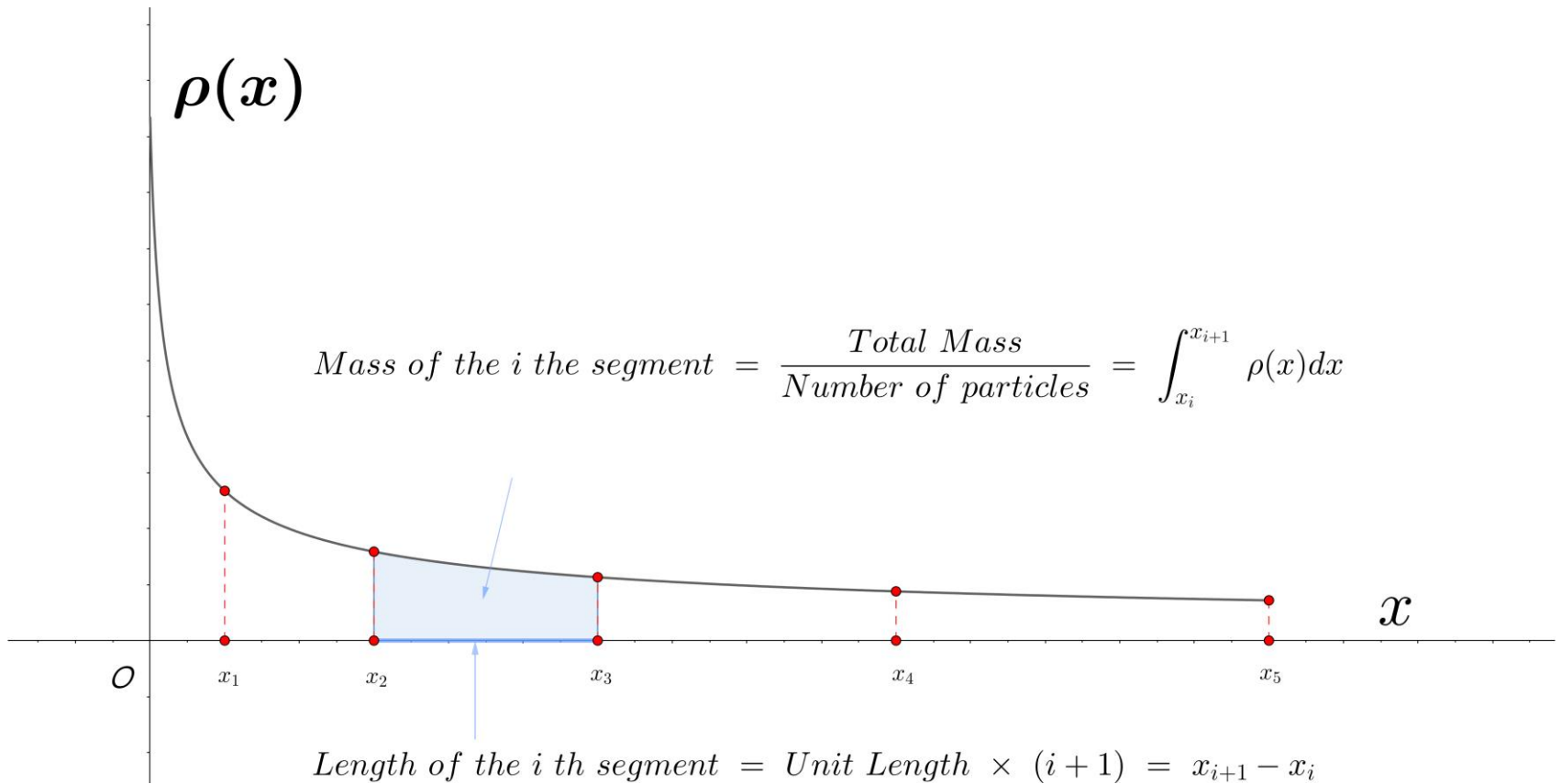
Fig. 16. Relation between the field equation and each segment

33

If the number of subsets is n and the length of the entire space is L, the position x_i of the sector dividing the specific region can be written as in Equation (3.5).

$$x_i = \sum_{k=1}^{i} l_k = \sum_{k=1}^{i} k \times l_1 = \frac{i(i+1)}{n(n+1)} \times L \tag{3.5}$$

if i is a positive integer value, Equation (3.6) holds.

$$\int_0^{x_i} \rho(x)dx = \int_0^{\frac{i(i+1)}{n(n+1)} \times L} \rho(x)dx = \frac{M}{n} \times i \tag{3.6}$$

Assuming that Equation (3.6) holds even when i is a positive real number, Equation (3.8) can be obtained by differentiating both sides by i in Equation (3.7).

$$\frac{L(2i+1)}{n(n+1)} \rho\left(\frac{i(i+1)}{n(n+1)} \times L\right) = \frac{M}{n} \tag{3.7}$$

$$\rho\left(\frac{i(i+1)}{n(n+1)} \times L\right) = \frac{M(n+1)}{L(2i+1)} \tag{3.8}$$

To express i as x, Equation (3.11) can be obtained by solving Equation (3.10) from Equation (3.9).

$$x = \frac{i(i+1)}{n(n+1)} \times L \tag{3.9}$$

$$i^2 + i - \frac{n(n+1)}{L}x = 0 \tag{3.10}$$

$$i = \frac{1}{2}\left(\sqrt{\frac{4n(n+1)}{L}x + 1} - 1\right) \tag{3.11}$$

Because i is assumed to be a positive real number, x_i can also be regarded as a positive real number with continuity. That is, for an arbitrary x, $\rho(x)$ can be expressed as Equation (3.12).

$$\rho(x) = \frac{M(n+1)}{L(2i+1)} = \frac{M(n+1)}{L\left(\sqrt{\frac{4n(n+1)}{L}x + 1}\right)} \tag{3.12}$$

$$\therefore \rho(x) \propto \frac{1}{x^{0.5}} \quad in \ one-dimensional \ case$$

Fig. 17 shows the result obtained through this field equation.
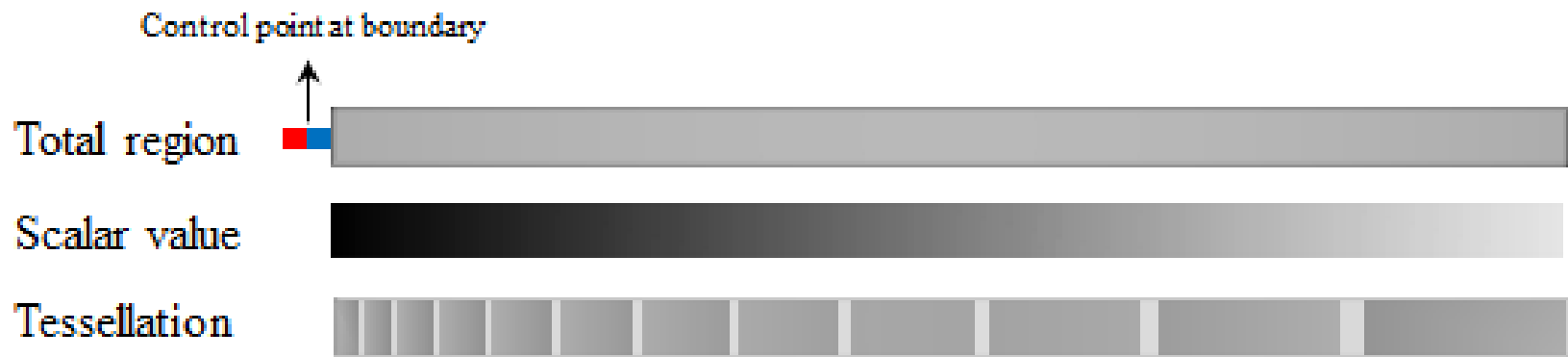
Fig. 17. An example result of the 1D case
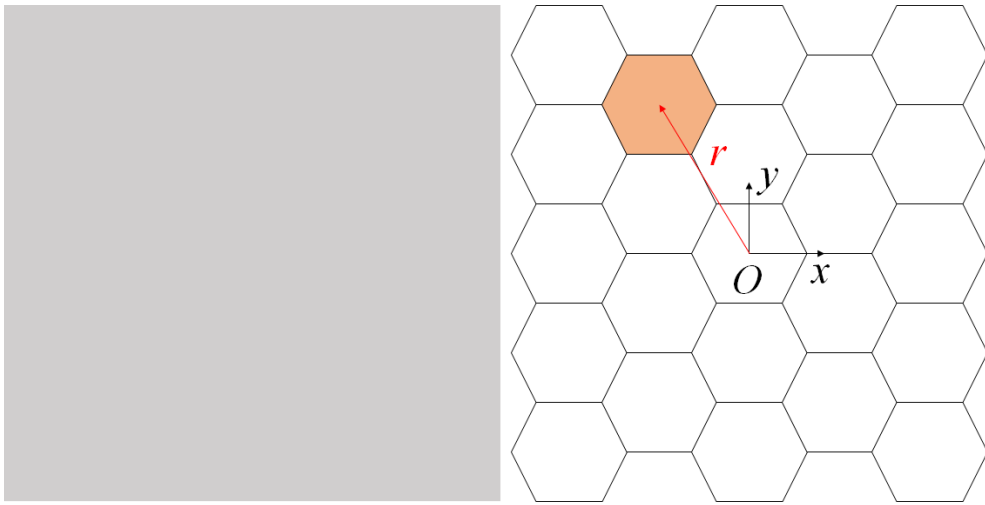
3.3.2. Two dimensional case

Fig. 18. 2D uniform scalar field (left),

expected tessellation result (right)

In the 2D case, unlike the 1D case, not only the size of the subsections but also their arrangement must be considered. In the simplest case, when the scalar field is uniform, a honeycomb structure can be expected, as shown in Fig. 18.
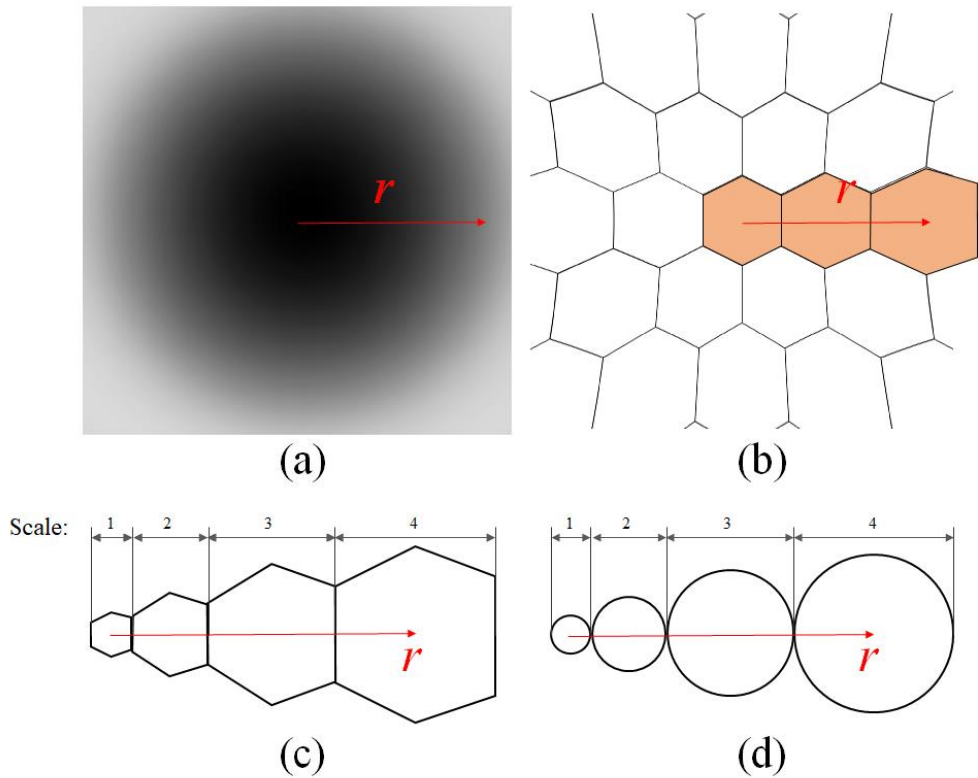
Fig. 19. An example of a 2D case, (a) visualization of a scalar field, (b) distorted honeycomb shapes, (c) distorted honeycomb shapes along radial direction, (d) circular approximation of (c)

However, it is difficult to derive the field equation corresponding to the case where the scale of the cell changes linearly as shown in Fig. 19 (b), (c) unlike the 1D case. In this study, the problem is simplified using the Euclidean distance r between the control point O and a specific location (x, y) as a parameter. In this case, the position (x, y) can be expressed in a polar coordinate system centered on the control point as shown in Equations (3.13) and (3.14).

38

$$\rho(x, y) = \rho(r), \tag{3.13}$$

where r is the distance between position (x, y) and control point (x_0, y_0)

$$r = \sqrt{(x - x_0)^2 + (y - y_0)^2} \tag{3.14}$$

Similar to the 1D case, assuming that the scale of each region increases linearly with the distance r, Equation (3.15) holds.

$$\int_{\theta_l(r)}^{\theta_h(r)} \int_{r_{i-1}}^{r_i} \rho(r) dr\, d\theta = constant, \quad for\ i = 1, 2, \dots, n \tag{3.15}$$

However, it is highly difficult to determine $\rho(r)$ analytically in Equation (3.15), because the integral range in the theta direction can vary with the distance r in the radial direction. The meaning of the integral form can be found through a 1D example. Equation (3.7) can be rewritten as in Equations (3.16) to (3.19).

$$\frac{L(2i + 1)}{n(n + 1)} \rho(x_i) = \frac{M}{n} \tag{3.16}$$

$$(i + 0.5) \frac{L}{n(n + 1)/2} \times \rho(x_i) = \frac{M}{n} \tag{3.17}$$

$$(i + 0.5)l_1 \times \rho(x_i) = \frac{M}{n} \qquad (3.18)$$

$$i \times l_1 \times \rho(x_i) \; + \; 0.5 \times l_1 \times \rho(x_i) = \frac{M}{n} \qquad (3.19)$$

Considering $\rho(x_i)$ as the representative value and $0.5 \times l_1 \times \rho(x_i)$ as the error term, Equation (3.19) can be thought of as follows.

$$Total\ area\ \times\ Representative\ value\ +\ error = Total\ scalar\ value$$

For the error term, the unit length l_1 is $O\left(\frac{1}{n^2}\right)$ and $\rho(x_i)$ is $O(1)$, so it can be considered $O\left(\frac{1}{n^2}\right)$. That is, if the number of cells n is large enough, this error can be acceptably ignored.

Similar to the 1D case, if the representative value of a specific location and the area of the corresponding region can be calculated, an approximate representation of $\rho(r)$ can be determined.

Actually, each cell would be created as a distorted hexagon as shown in Fig. 19 (c). However, as already mentioned, it is highly difficult to calculate those areas analytically. Therefore, the shape of each cell is approximated to a circle with linearly increasing scaling factor. The

positions of the sectors dividing each cell are triangular numbers as shown in Equation (3.20).

$$r_i = \sum_{k=1}^{i} s_k = \sum_{k=1}^{i} k \times s_1 = \frac{i(i+1)}{2} \times s_1 \; , \tag{3.20}$$

$$where, s_i \; is \; a \; diameter \; of \; the \; i^{th} \; region$$

Assuming that r_i and i are positive real values, Equation (3.21) holds.

$$r = \frac{i(i+1)}{2} \times s_1 \tag{3.21}$$

Equation (3.21) can be rewritten in the form of an implicit function, as Equation (3.22). Then, i can be expressed for r as in equation (3.23).

$$i^2 + i - \frac{2r}{s_1} = 0 \tag{3.22}$$

$$i = \frac{1}{2}\left(\sqrt{\frac{8r}{s_1} + 1} - 1 \right) \tag{3.23}$$

If the representative value of each cell is set as the center value of

41

the circle, Equation (3.24) holds.

$$\frac{\pi s_i^2}{4}\rho(r) = \frac{M}{n} \tag{3.24}$$

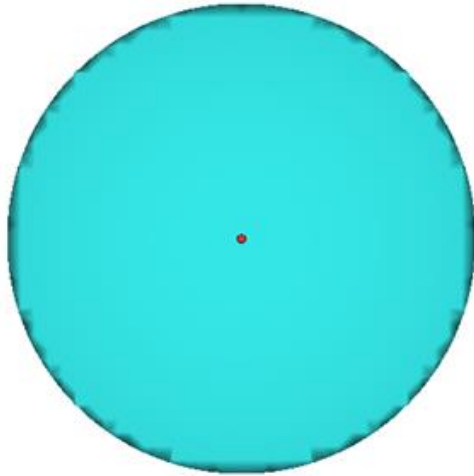Equation (3.24) can be rewritten as Equations (3.25) and (3.26).

$$\rho(r) = \frac{M}{n}\frac{4}{\pi i^2 s_1^2} \tag{3.25}$$

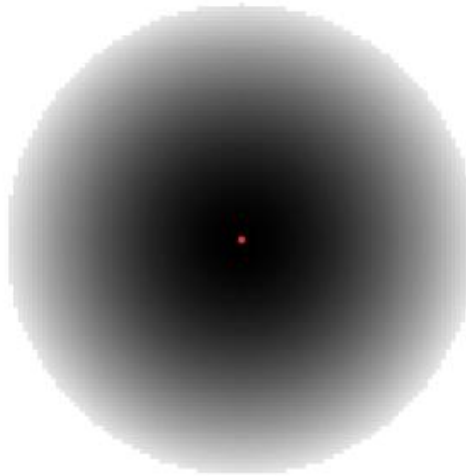$$\rho(r) = \frac{16M}{\pi n s_1^2\left(\sqrt{\frac{8r}{s_1}+1}-1\right)^2} \tag{3.26}$$

$$\therefore \rho(r) \propto \frac{1}{r^{1.0}} \quad in \ two-dimensional \ case$$

Fig. 20 shows the tessellation result through this field equation $\rho(r)$.

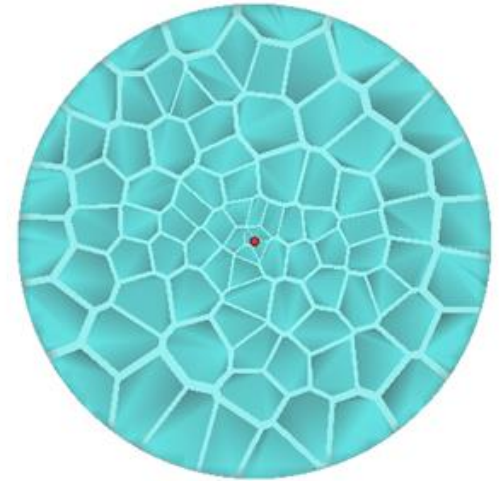Total region   Scalar field   Tessellation

Fig. 20. An example of a 2D case

3.3.3. Three-dimensional case

For a 3D case, similar to the case for 2D, it is assumed that the shape of each cell is a sphere and the scale factor increases linearly in the radial direction. As shown in Equation (3.27), the field equation $\rho(x, y, z)$ is expressed as $\rho(r)$ having only the radial distance r as a parameter.

$$\rho(x, y, z) = \rho(r),$$

(3.27)

where r is the distance between position (x, y, z) and control point (x_0, y_0, z_0)

The positions of the sectors dividing each cell are triangular numbers as shown in Equation (3.28).

$$r_i = \sum_{k=1}^{i} s_k = \sum_{k=1}^{i} k \times s_1 = \frac{i(i+1)}{2} \times s_1$$

(3.28)

where, s_i is a diameter of the ith region

Assuming that r_i and i are positive real values, Equation (3.29) holds.

$$r = \frac{i(i+1)}{2} \times s_1$$

(3.29)

Equation (3.29) can be rewritten as Equations (3.30) and (3.31).

$$i^2 + i - \frac{2r}{s_1} = 0 \qquad (3.30)$$

$$i = \frac{1}{2}\left(\sqrt{\frac{8r}{s_1} + 1} - 1\right) \qquad (3.31)$$

If the representative value of each cell is set as the center value of the circle, Equation (3.32) holds.

$$\frac{4\pi s_i^3}{3}\rho(r) = \frac{M}{n} \qquad (3.32)$$

Equation (3.32) can be rewritten as Equations (3.33) and (3.34).

$$\rho(r) = \frac{M}{n}\frac{4}{\pi i^3 s_1^3} \qquad (3.33)$$

$$\rho(r) = \frac{32M}{\pi n s_1^3 \left(\sqrt{\frac{8r}{s_1} + 1} - 1\right)^3} \qquad (3.34)$$

$$\therefore \rho(r) \propto \frac{1}{r^{1.5}} \ \ in \ three-dimensional \ case$$

45

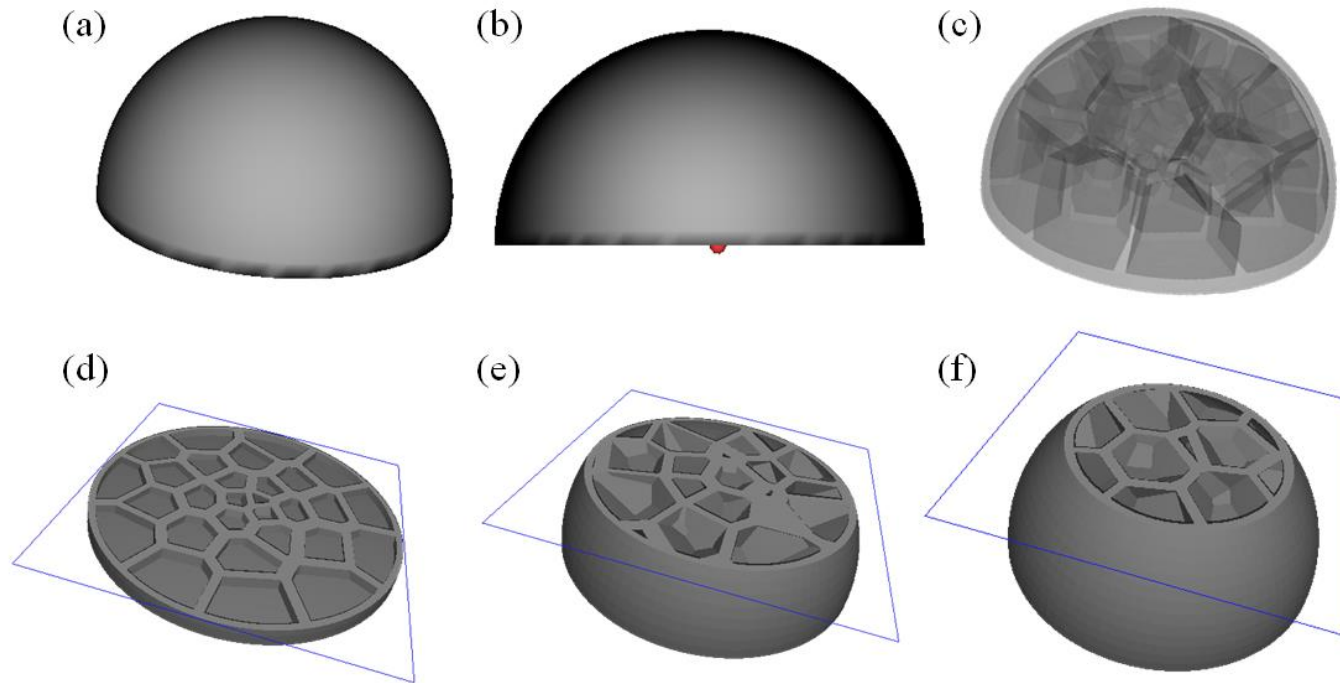Fig. 21 shows the tessellation result through this field equation $\rho(r)$.

Fig. 21. An example of a 3D case, (a) input surface mesh, (b) experiment environment, (c) tessellated 3D Voronoi structure, (d, e, f) visualization of the clipped volume.
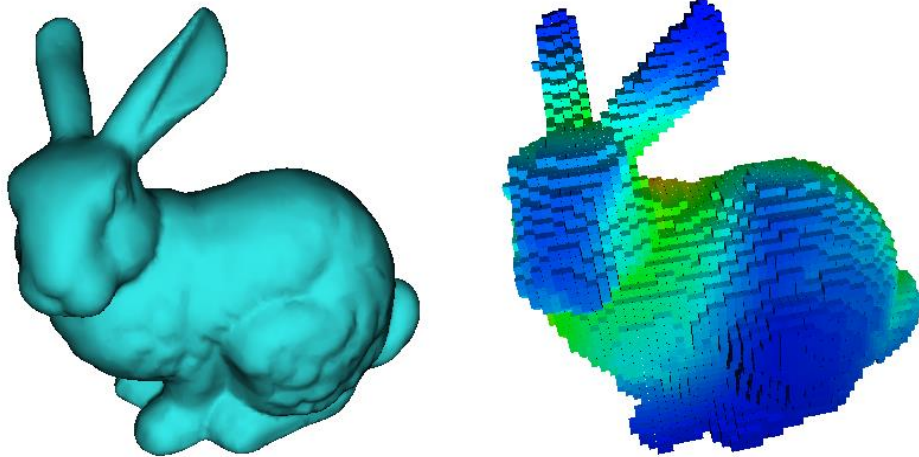
3.4. Implementation

Fig. 22. Input surface mesh (left) and its solid voxelization (right)

It is possible to calculate the continuous field values through the equations derived in section 3.3, but it is difficult to analytically integrate the total field values of each Voronoi cell. Therefore, in this research, the numerical integration method is used.

First, the input solid model is voxelized as shown in Fig. 22. Then each voxel is clustered to the nearest Voronoi cell, followed by the numerical integration being performed through the sum of each voxel values as shown in Fig. 23.

Each voxel value is calculated independently for each control point. If there are multiple control points, the largest value is determined as the voxel value. This is because if the voxel value is calculated as an overlap sum according to the control points, unlike the intuition

initially thought, dense inner structure may appear in areas further away from control points.
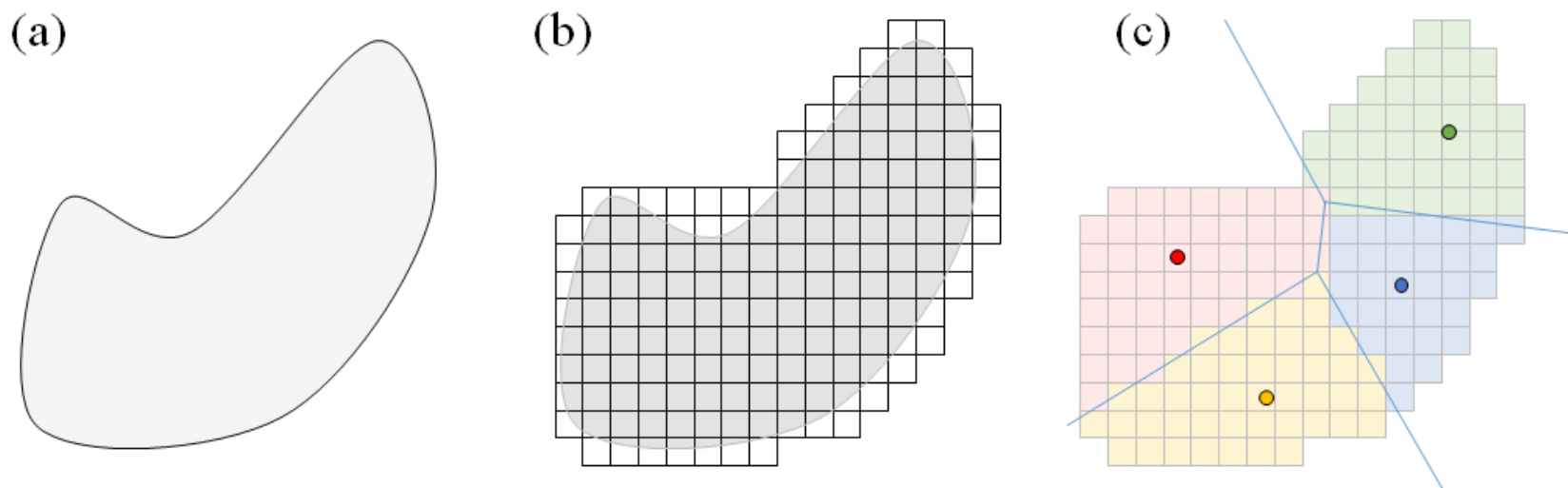
Fig. 23. (a) input surface mesh, (b) solid voxelization of (a), (c) Voronoi cells and their dependent voxels

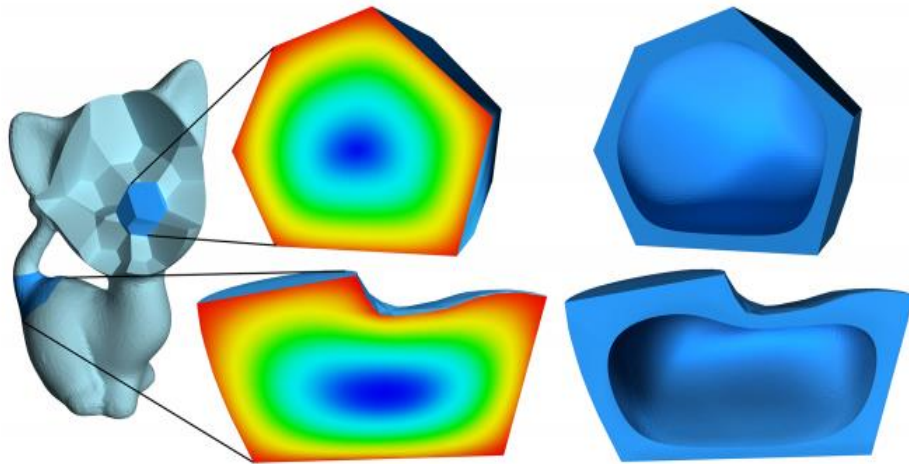## 3.5. Creation of the volumetric cell structure

Fig. 24. Harmonic porous carving [30]

It is required for fabrication to make each Voronoi cell has a skin of a certain thickness with void inside as shown in Fig. 24. Lin Lu et al. [30] calculated the harmonic field of each cell to obtain the inner iso-surface of each; the volumetric porous structure was acquired as shown in Fig. 24. However, in this method, considerable computational power is required because the finite element method of each cell should be solved. Therefore, in this study, cellular carving is performed through volume-based scaling and Catmull-Clark subdivision algorithm is adapted to obtain similar results compared to Lin Lu et al.'s.

3.5.1 Volume-based scaling

The empty space of a porous structure can be represented by the inner surfaces generated using the outer surface of each cell. That is, the space between the outer and inner surfaces of each cell will be filled with the materials during the additive manufacturing process. Therefore, the scaling factor s_f of each cell determines the wall thickness of the generated porous structure. By referring to Zhuo Wei's work [53], the equation to calculate the scaling factor is determined as in Equation (3.35), while V_c is the volume of the original cell, and V_c' is the volume to be left.

$$s_f = \sqrt[3]{\frac{V_c - V_c'}{V_c}} \qquad (3.35)$$

Because it is tedious work to determine V_c' for each cell individually, a type of volume distribution method is used. The designer uses the entire volume to leave V' as an input parameter. Additionally, it can be said that the distribution method determines the behavior of the outer wall thickness. In this paper, the uniform distribution method is adapted as shown in Equation (3.36).

$$V_c' = \frac{V'}{Total\ number\ of\ cells} \qquad (3.36)$$

The maximum and minimum values of the scaling factor are set for exception handling. The default scaling factor range is $0.7-0.9$, but the designer can handle this range. Fig. 26 (a) shows the designed inner structure.

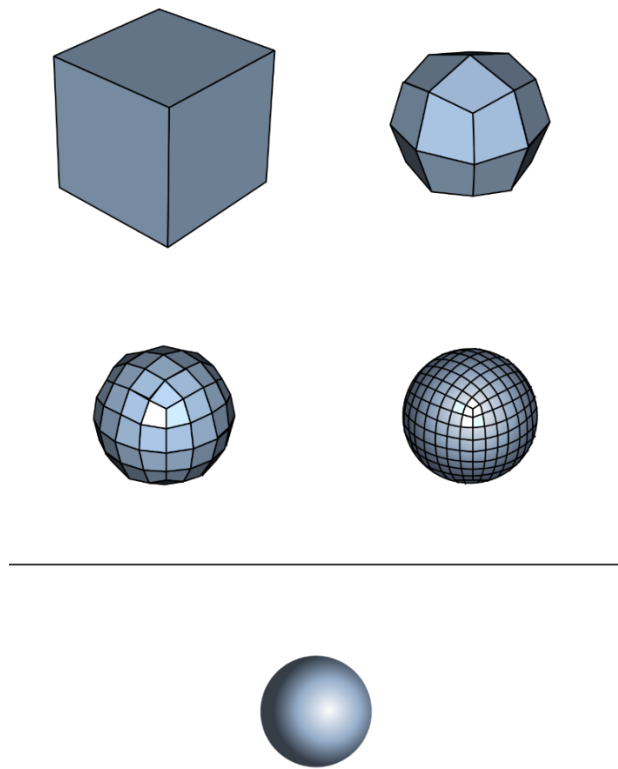## 3.5.2 Catmull-Clark Subdivision

Fig. 25. Catmull－Clark subdivision of a cube

The Catmull－Clark subdivision [54] is a method of creating smooth mesh surfaces as shown in Fig. 25. This method has the advantage for creating a smooth mesh quickly, but the result is shrunken compared to the original mesh. However, this shortcoming is acceptable in this paper. The shrunken region in the Catmull－Clark subdivision is the edge of the Voronoi cell and this region can play the same role as the fillet. Fig. 26 (b) shows the result after the Catmull－Clark subdivision process.
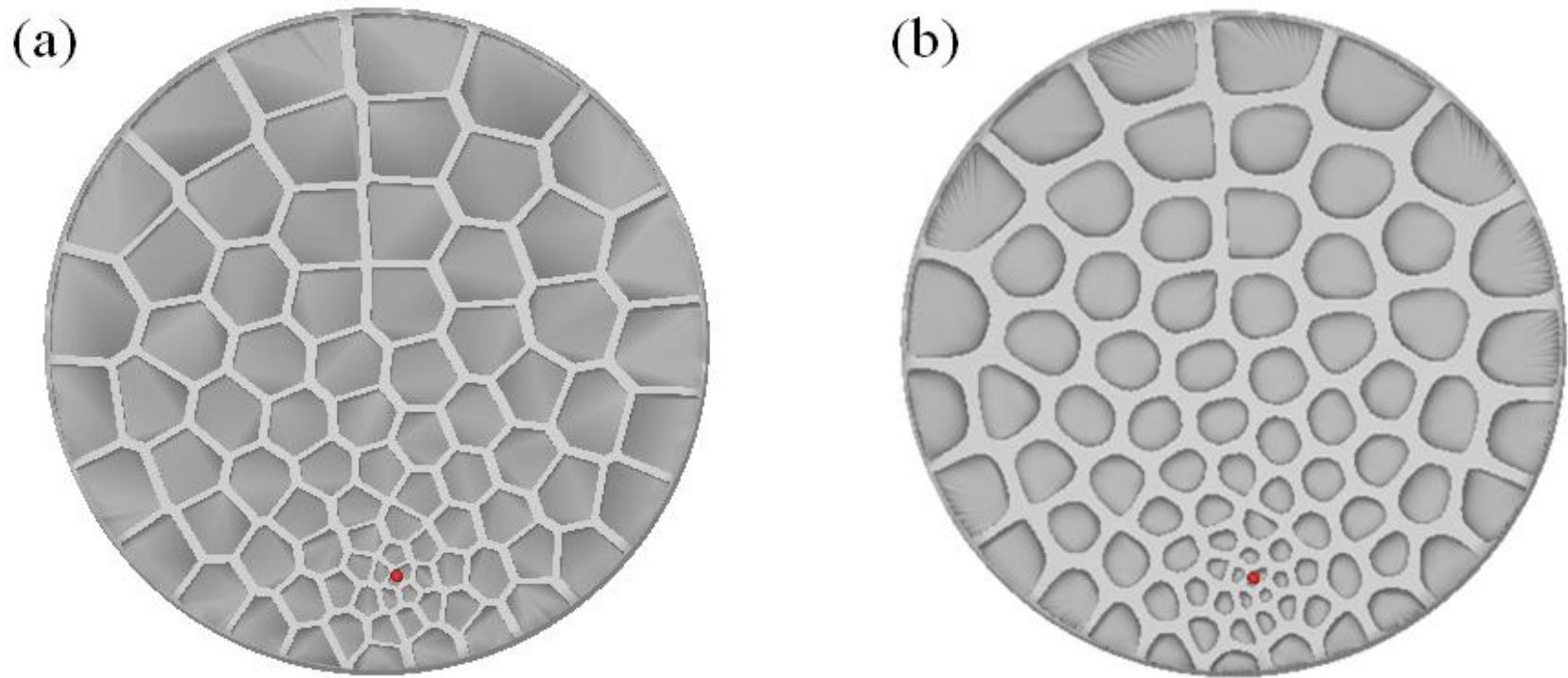
Fig. 26. (a) Initial Voronoi structure, (b) after Catmull−Clark subdivision.

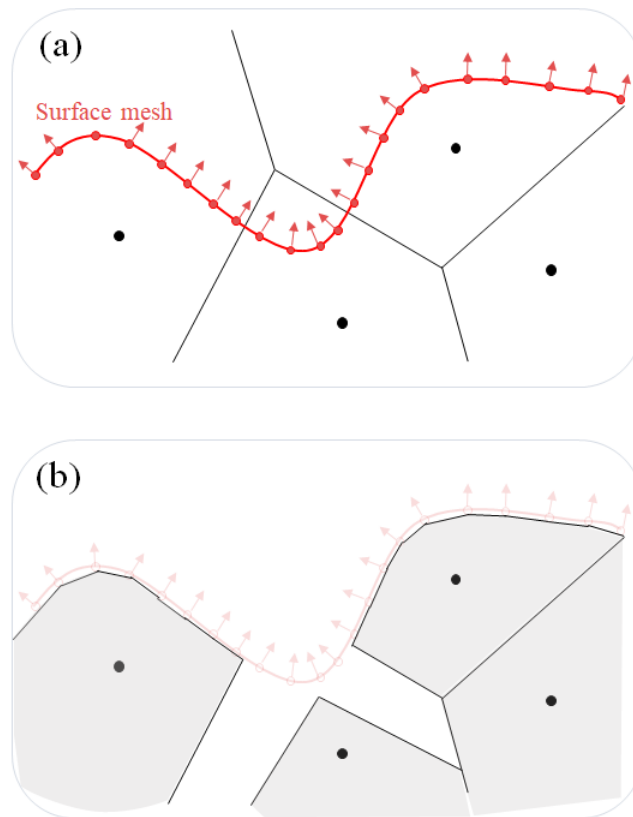### 3.5.3. On the boundary Voronoi cell

Fig. 27. Voronoi cell clipping, (a) before clipping, (b) after clipping

As mentioned earlier, because Voronoi tessellation is performed using bisectors between close points, the outermost Voronoi cells are open structures. Therefore, postprocessing is required to make the boundary Voronoi cell inside the input solid model. In this paper, the Voronoi clipping process [55] is performed as a postprocessing as shown in Fig. 27.

Chapter 4. Results and Discussion

4.1. Results

The program developed in this study is shown in Table 1 and from Fig. 28 to Fig. 39. The field equation used in this program is inductively determined via section 3.3. Specifically, Equation (4.1) is used, which takes the distance r between a voxel position and a control point as a parameter.

$$\rho(r) = \frac{1}{\left(\sqrt{r+1}-1\right)^c} \tag{4.1}$$

There are three parameters that the user can control: the strength of the control point c, the number of cells n, and the carving volume percentage p. The subdivision is the process of smoothing out the inner structure, so it has been implemented as a user−selectable toggle type.

Table 1 shows the operations used in the program. Fig. 28 shows the display when the program is initially run. When a solid model is loaded, it is displayed in the center of GUI as shown in Fig. 29. As shown in Fig. 30, when the user enters the voxel resolution, the program automatically performs a solid voxelization process as shown in Fig. 31. After that, the user adds any number of control points as in Fig.

32 and then enters each parameter as shown in Fig. 33. Fig. 34 through Fig. 39 show examples of designed inner structures.

Users may want to leave the area near the joints of the bracket as solid state, as shown in Fig. 40 (a). Because considerable effort is required to apply the parametric interface used in the existing CAD system, the program is configured to obtain the same result as if the constraint was adapted by using the original solid body and the modified solid body. First, the control points are set on the modified solid body as shown in Fig. 41. The designed inner structure is shown in Fig. 42. Then, as shown in Fig. 43 and Fig. 44, if the outer surface is replaced with the modified surface mesh, the constrained design can be obtained.

To measure the running time of the program, the same procedure was conducted for several examples. The program was executed on Windows 10 64bit, Intel i7-4930K CPU. The whole process was done on a single CPU, and the field values were calculated whenever a control point was changed or added. The tested models are shown in Fig. 45, and the statistics are shown in Table 2. The field value calculation time is excluded from the statistics because it is short enough for the user to interact with the program, and it increases linearly with the number of control points.

Program actions					
Group	Icon	Description	Group	Icon	Description
Create		Voxelize mesh	Camera Control		Isometric view
		Create inner structure			Top view
Rendering		Render outer mesh			Bottom view
		Render voxels			Front view
		Transparent rendering (inner structure)			Back view
Control Mode		Default (view control)			Left view
		Add control point			Right view
		Show clipped mesh			

Table 1. Control actions in the proposed program

59
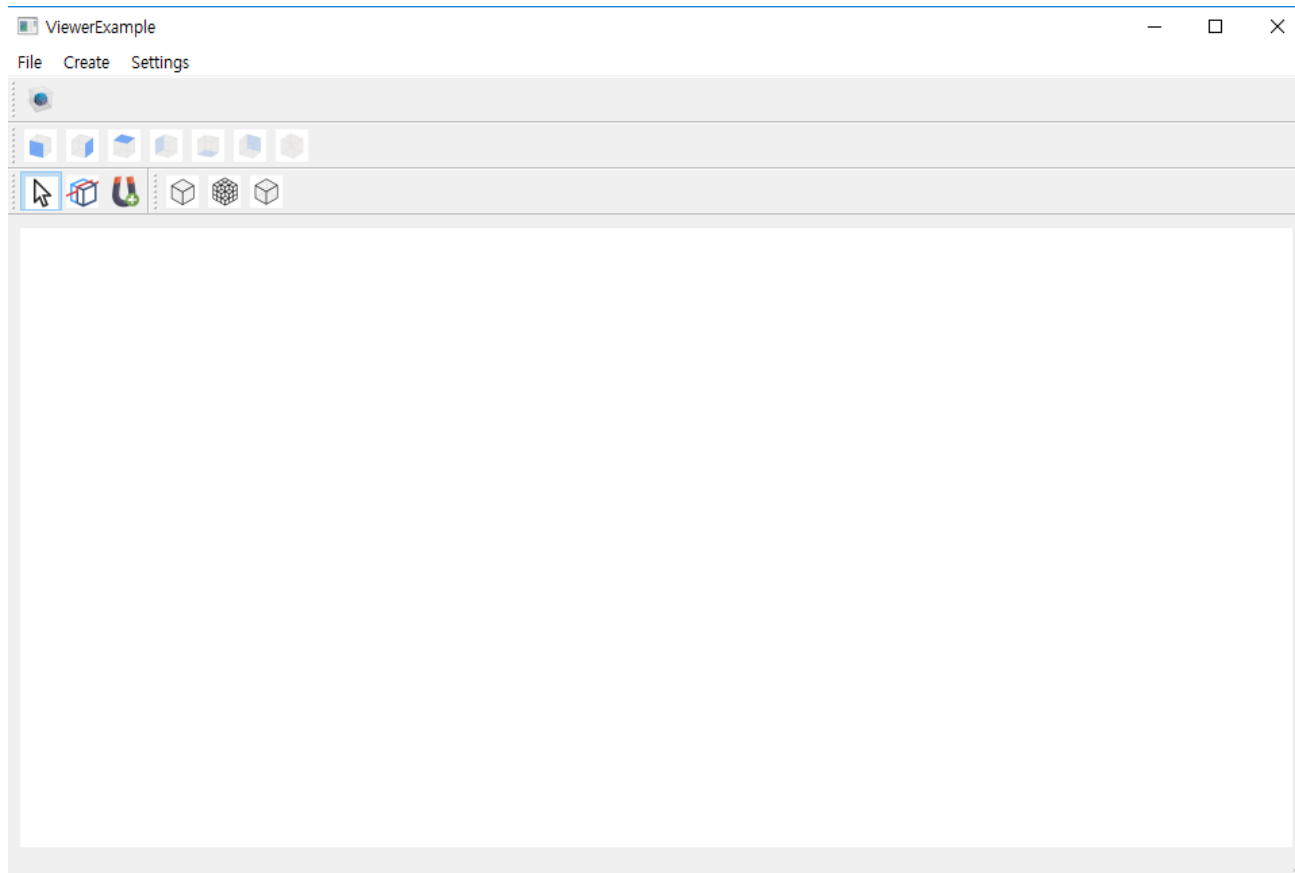
Fig. 28. Initial display
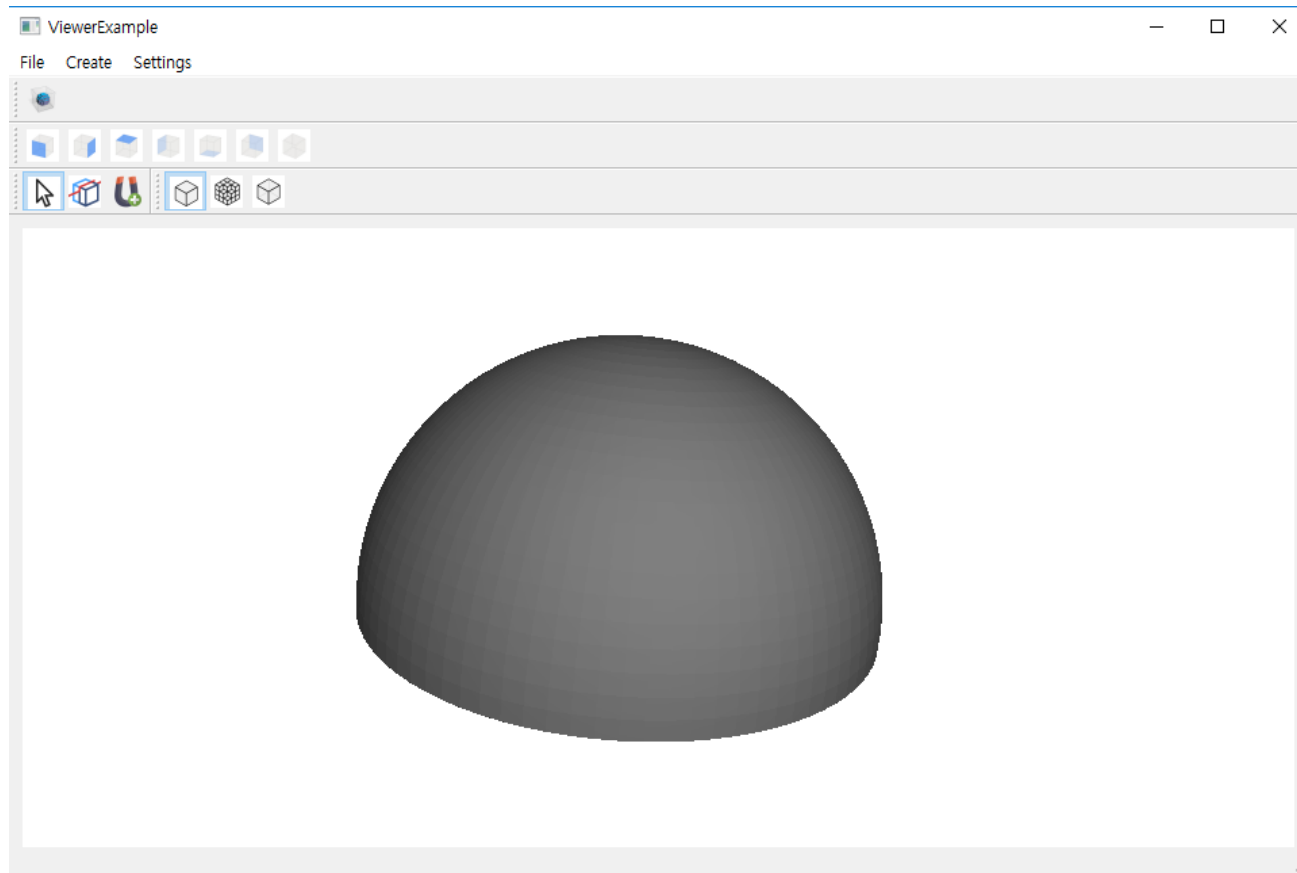
Fig. 29. Loaded mesh file (half sphere)
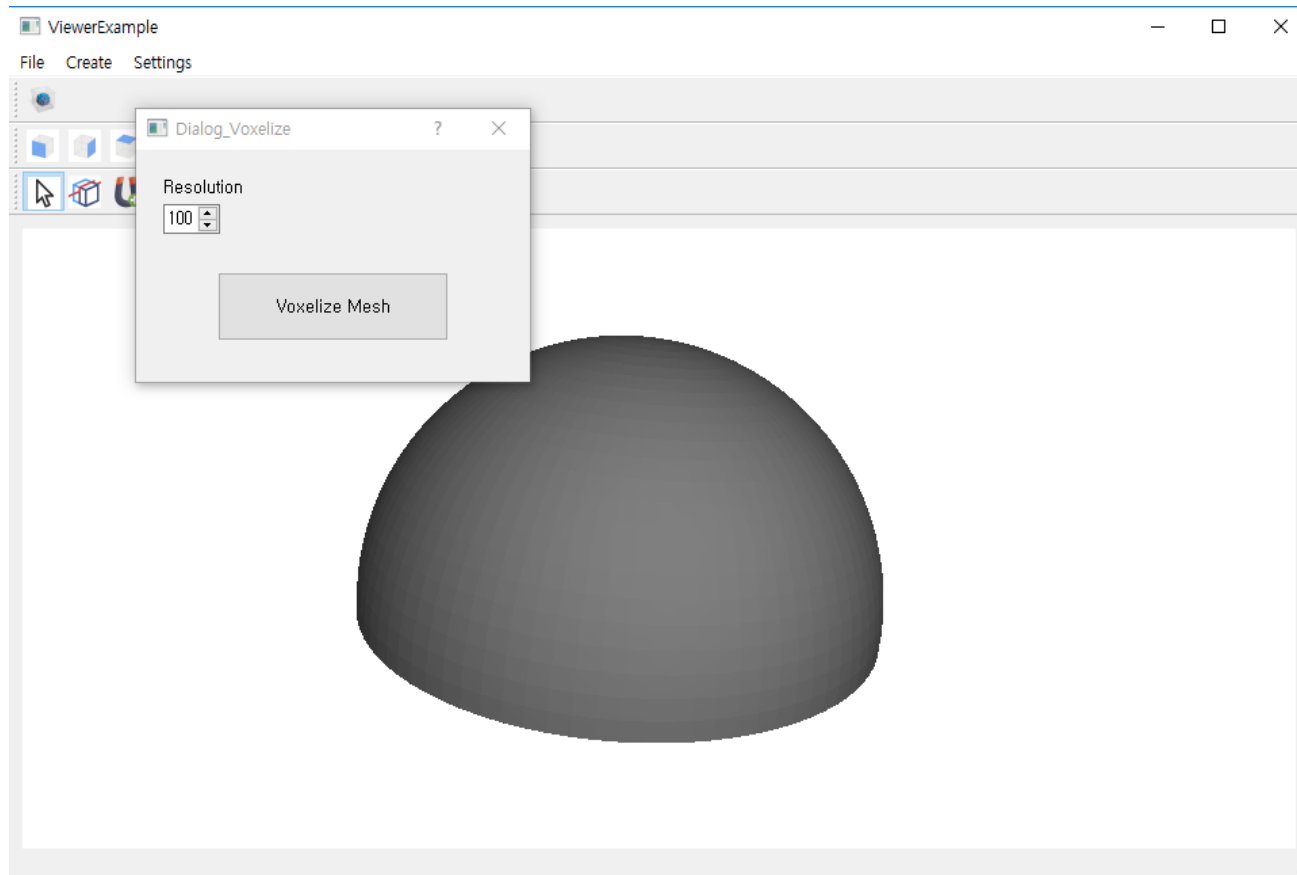
Fig. 30. Dialog for voxelization
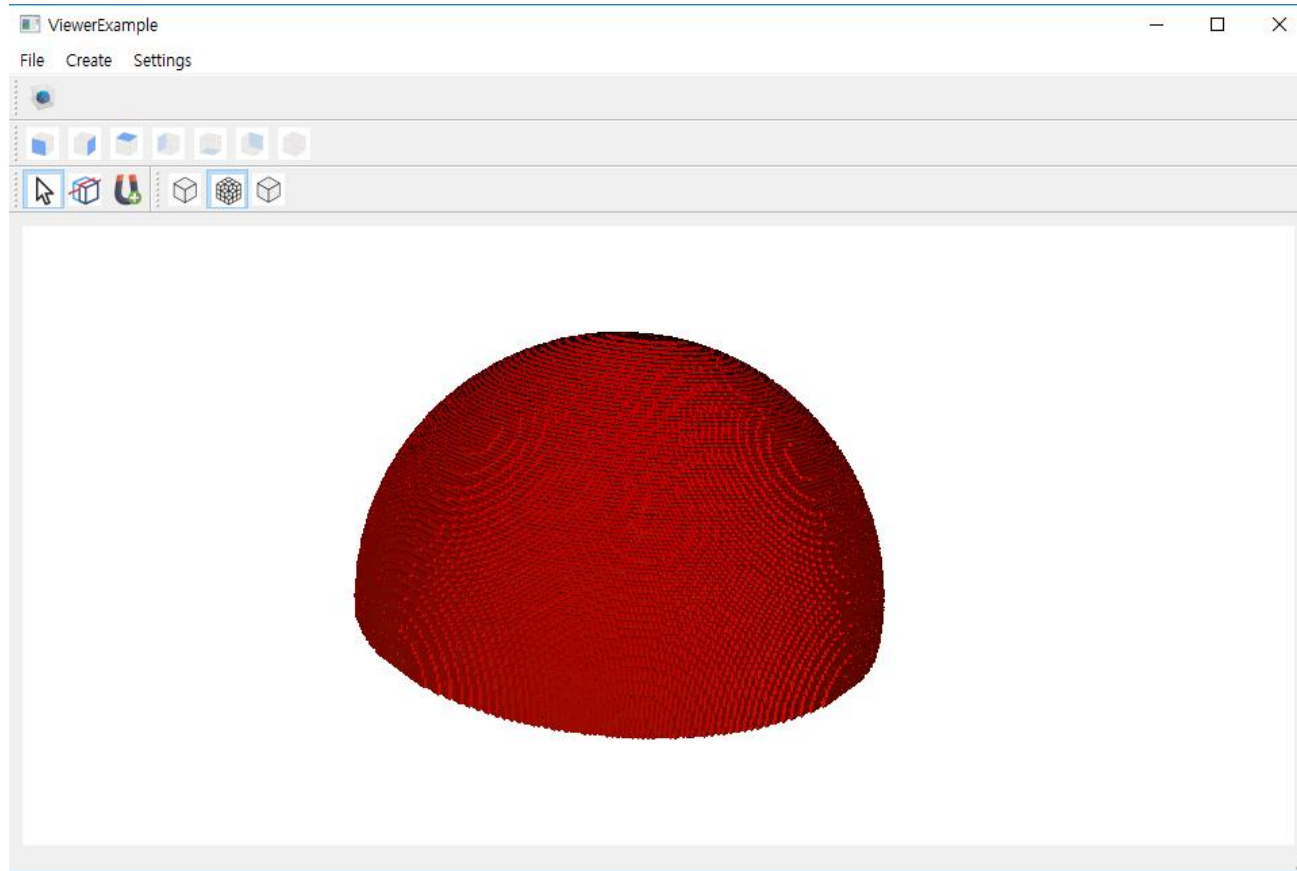
Fig. 31. Visualization of the voxelized mesh
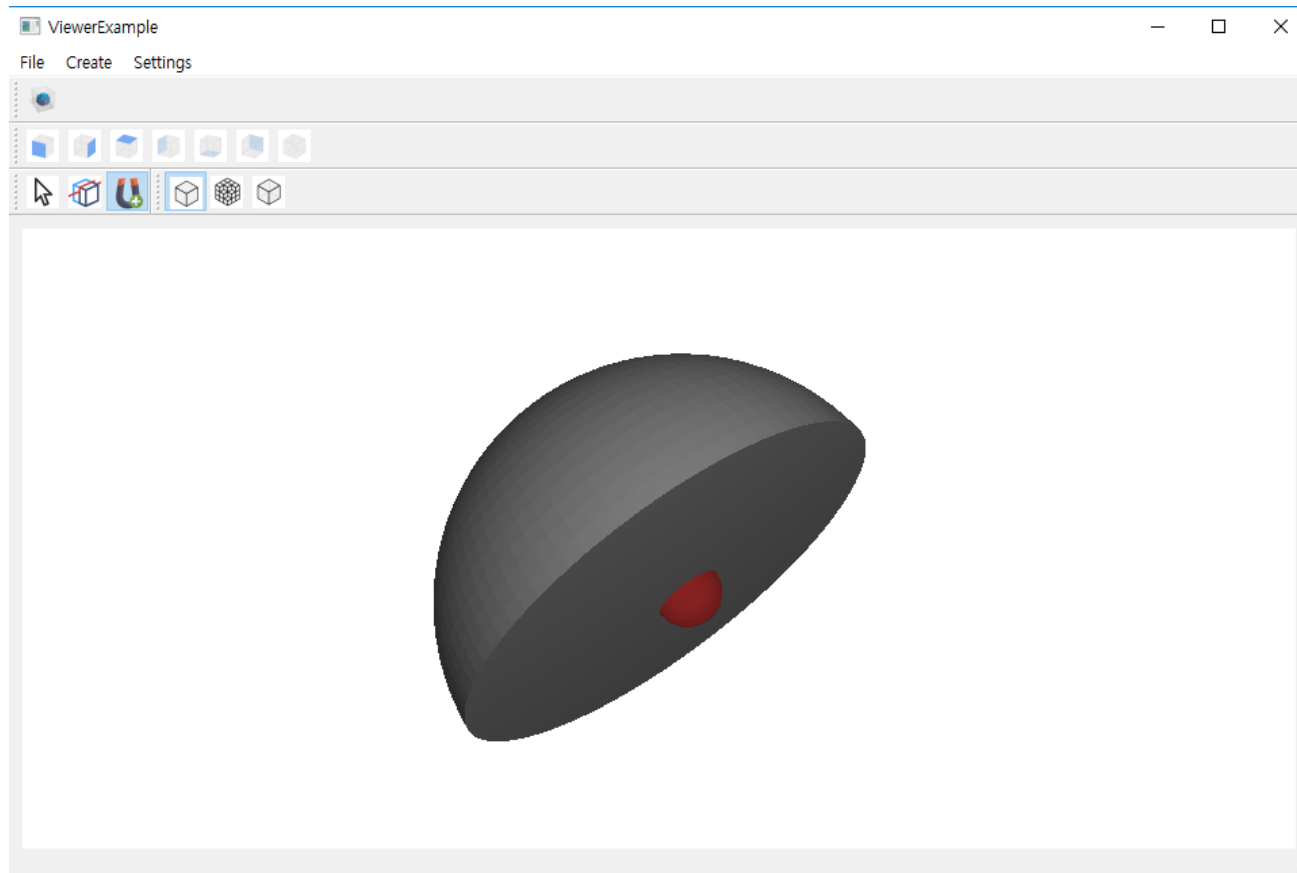
Fig. 32. A control point at the bottom of the mesh
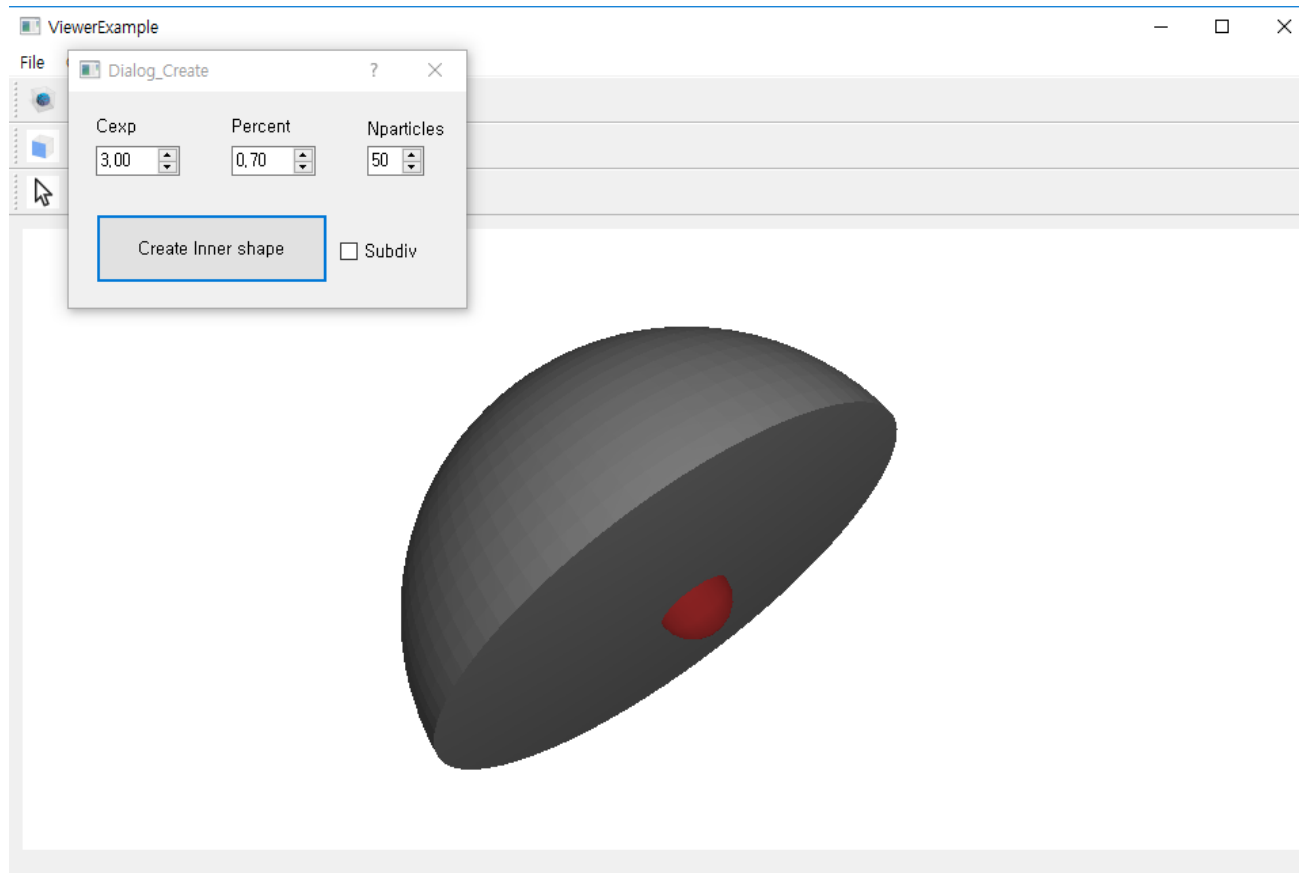
Fig. 33. Dialog for creating inner structure
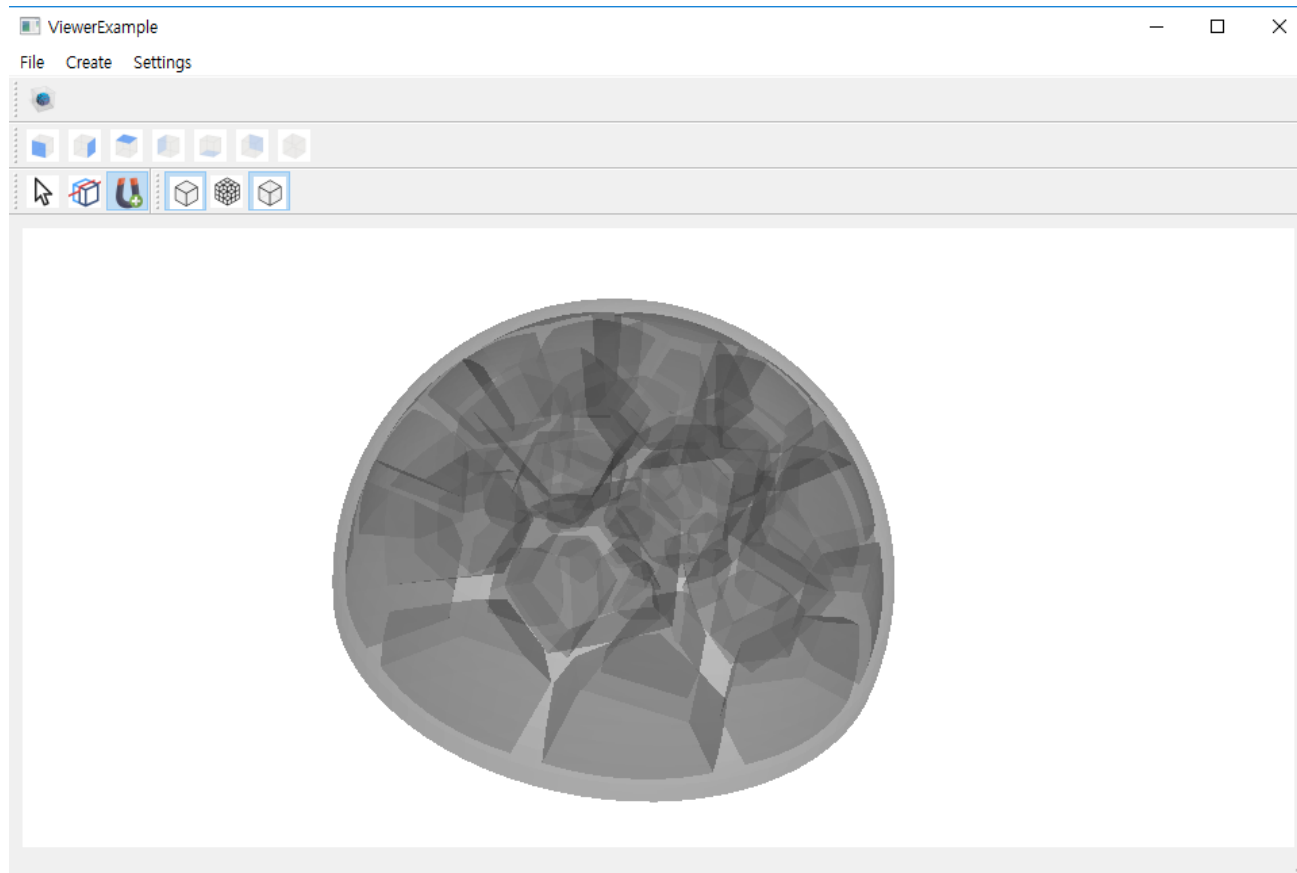
Fig. 34. Transparent visualization of the created inner structure before Catmull−Clark subdivision
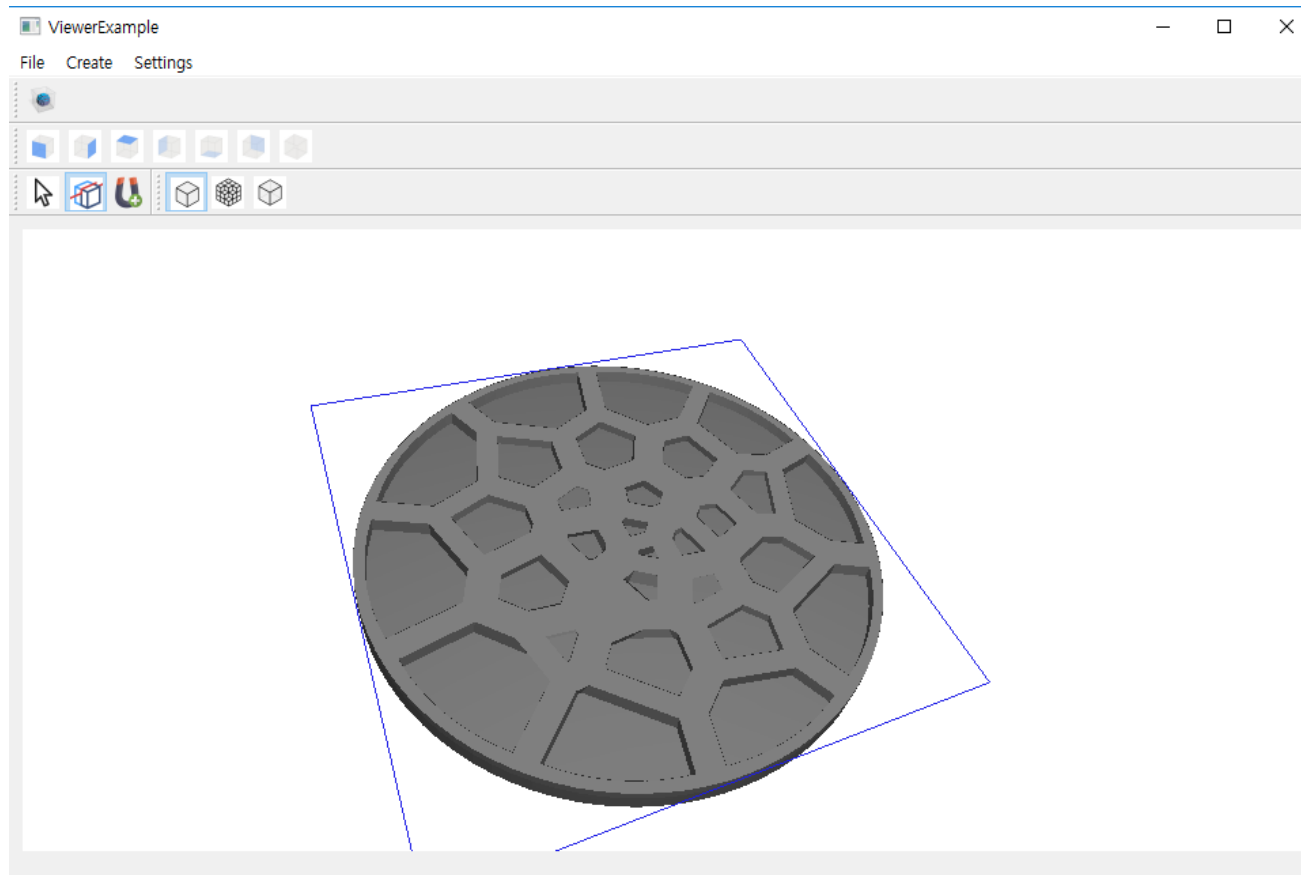
Fig. 35. Clipped visualization of the created inner structure before Catmull−Clark subdivision (bottom)
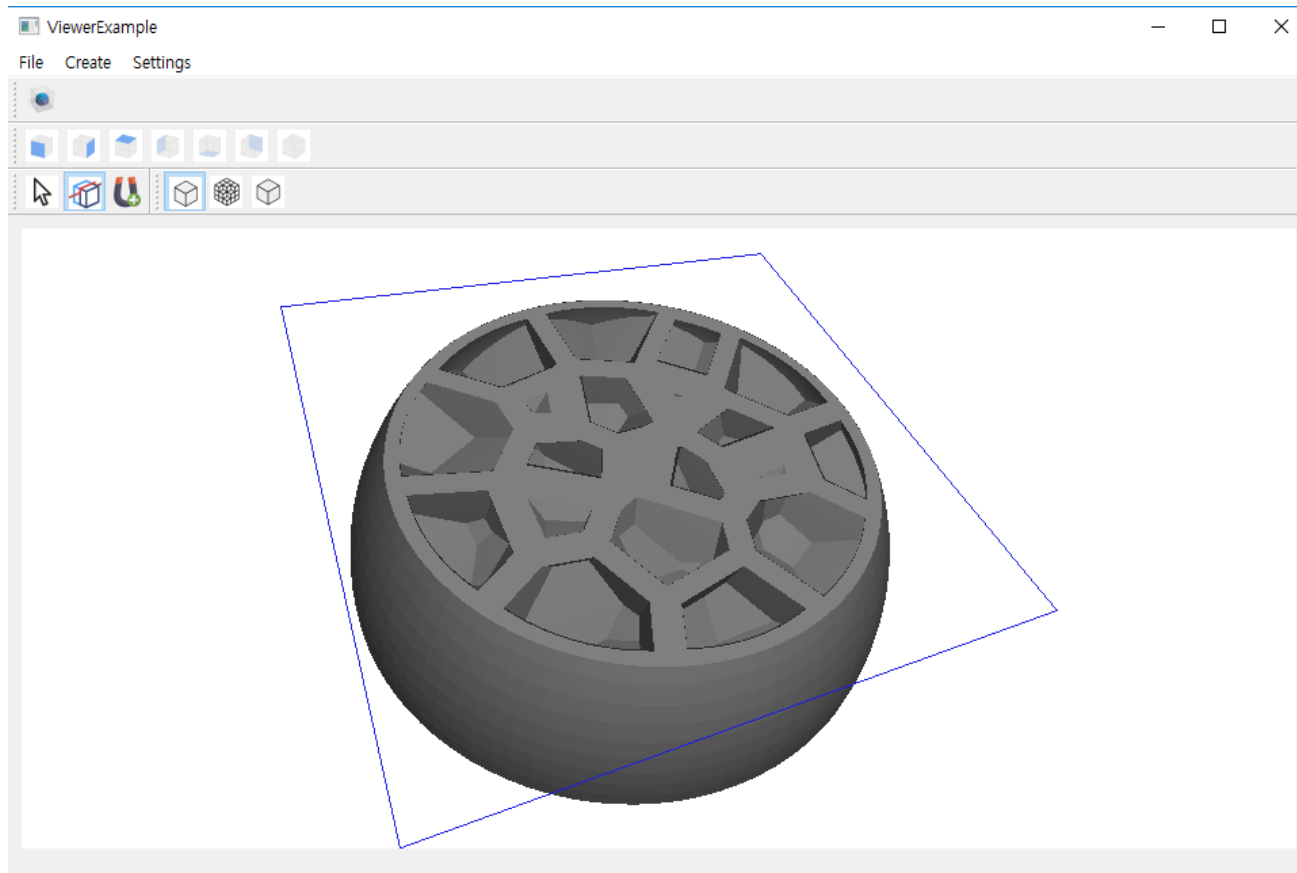
Fig. 36. Clipped visualization of the created inner structure before Catmull−Clark subdivision (center)
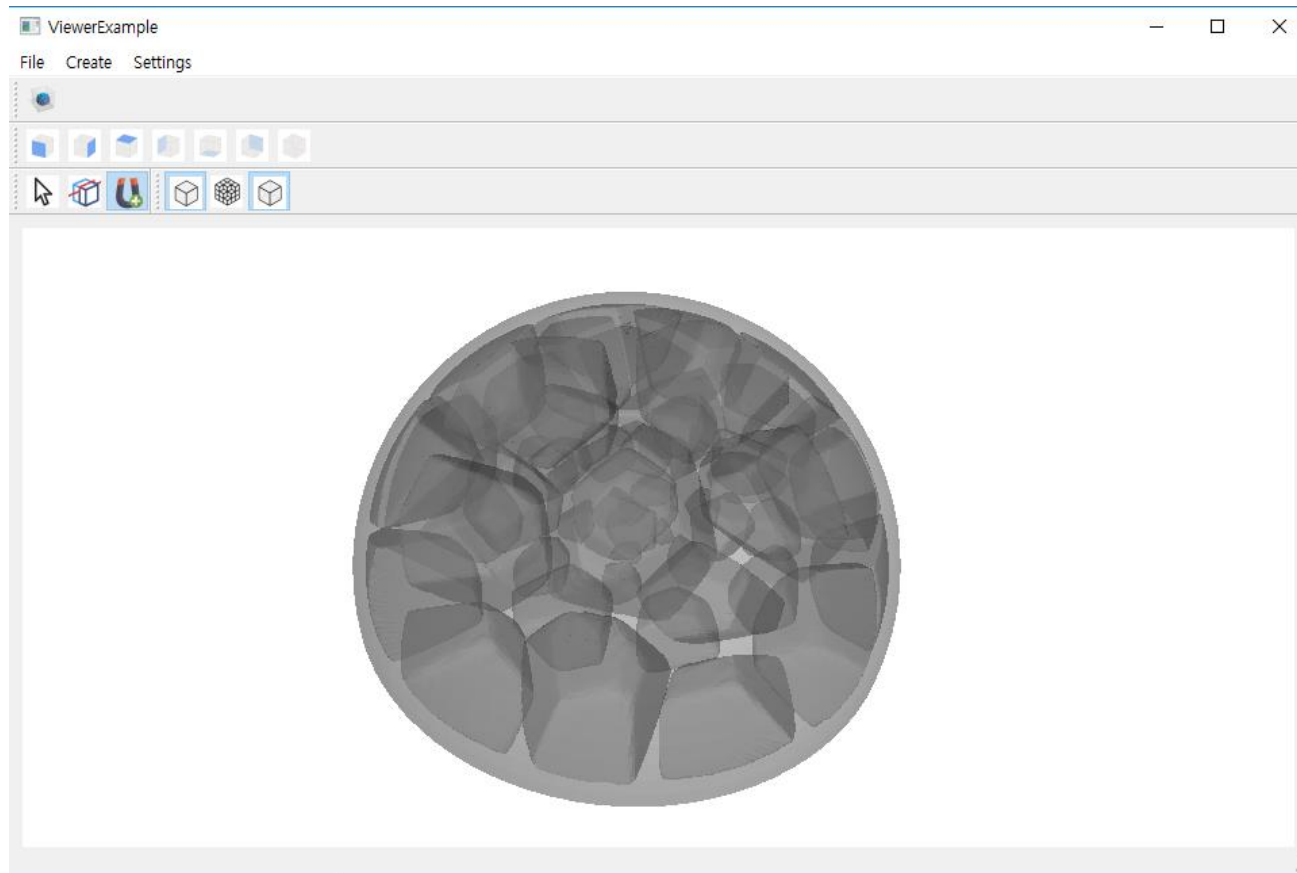
Fig. 37. Transparent visualization of the created inner structure after Catmull−Clark subdivision
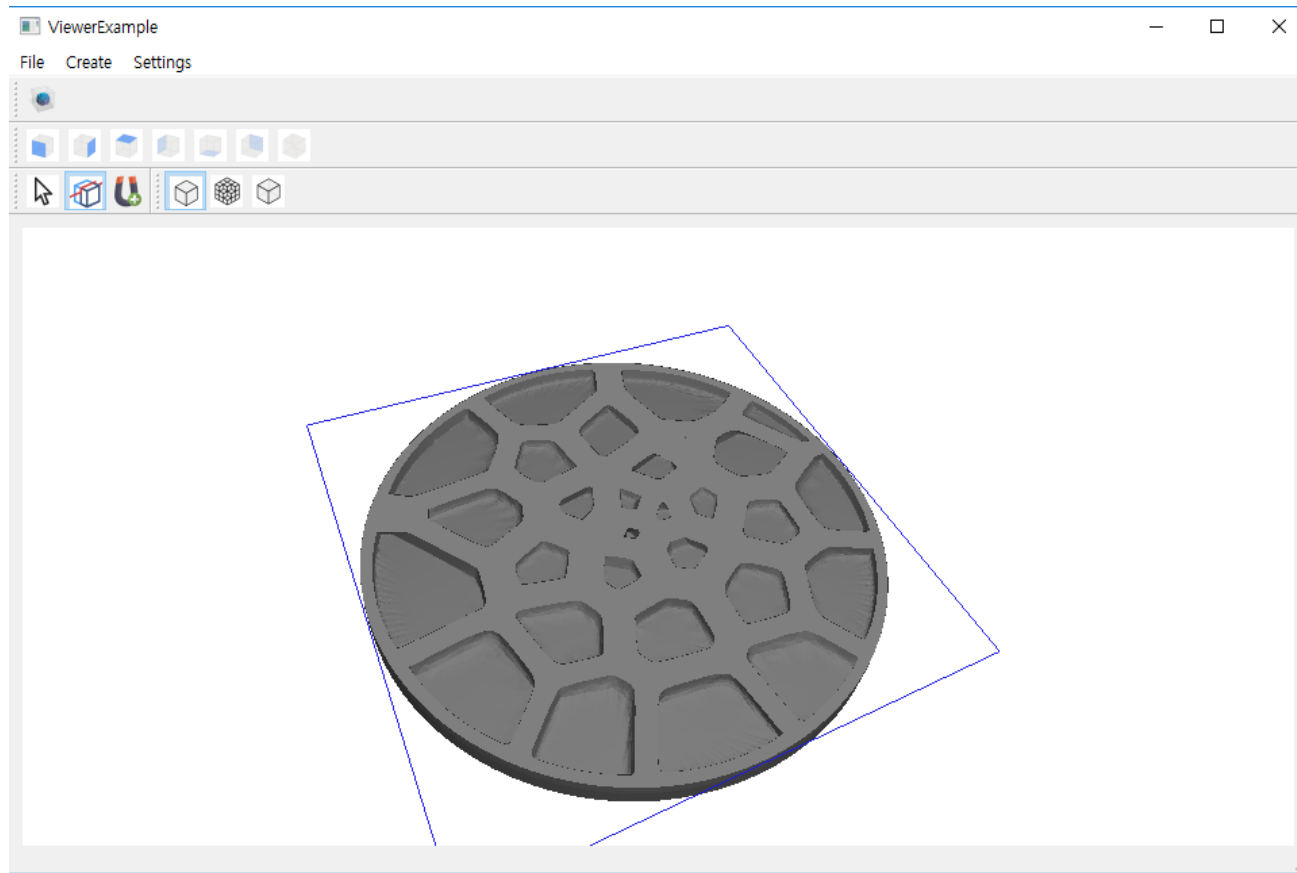
Fig. 38. Clipped visualization of the created inner structure after Catmull−Clark subdivision (bottom)
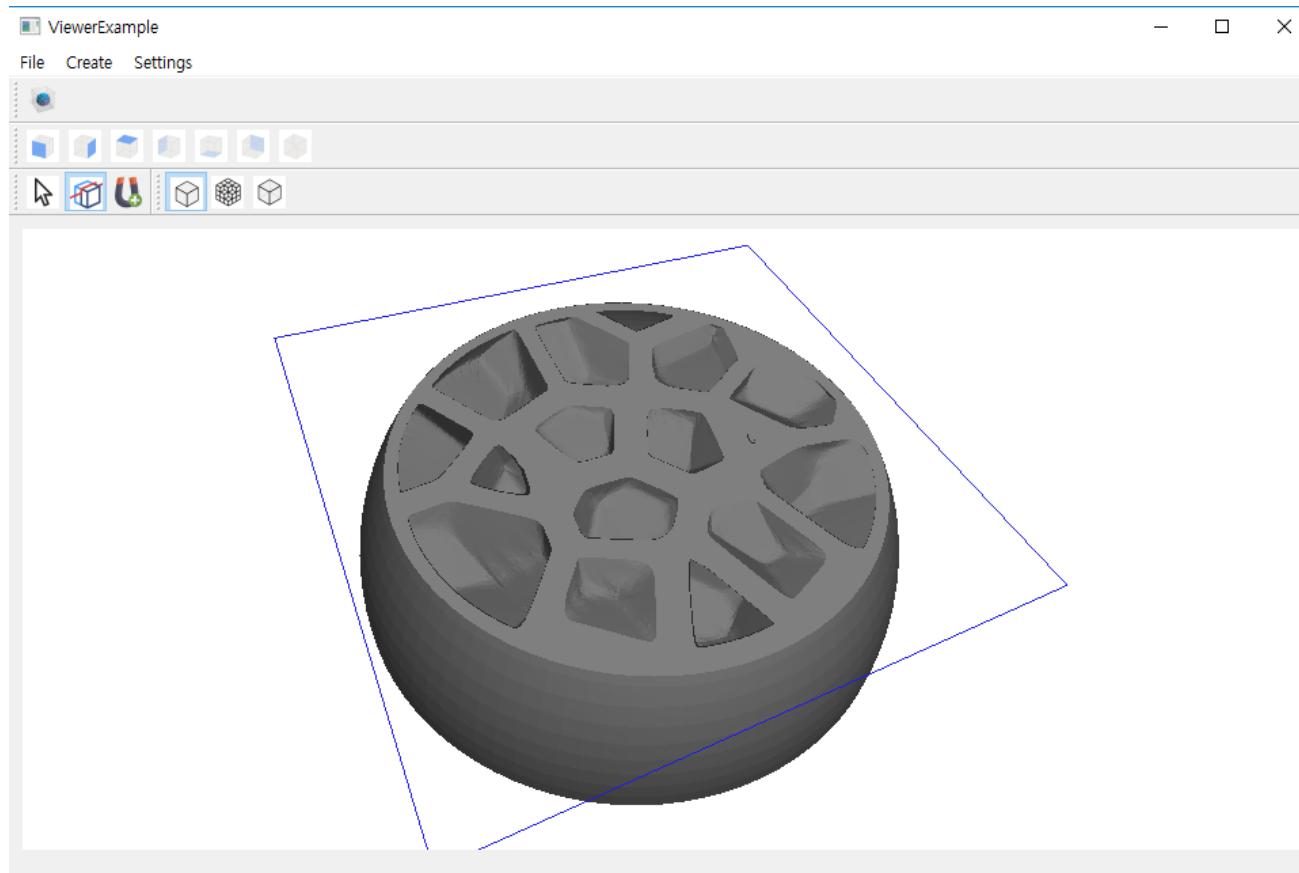
Fig. 39. Clipped visualization of the created inner structure after Catmull−Clark subdivision (center)
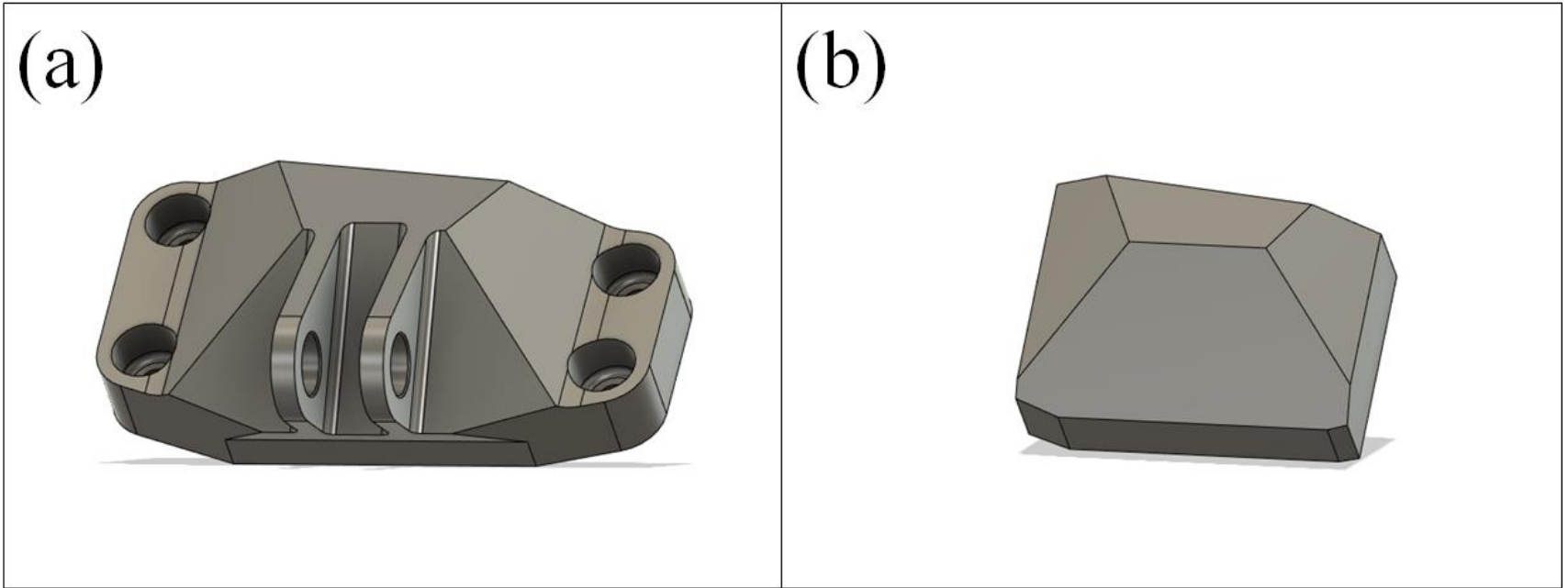
Fig. 40. (a) Original bracket, (b) edited bracket
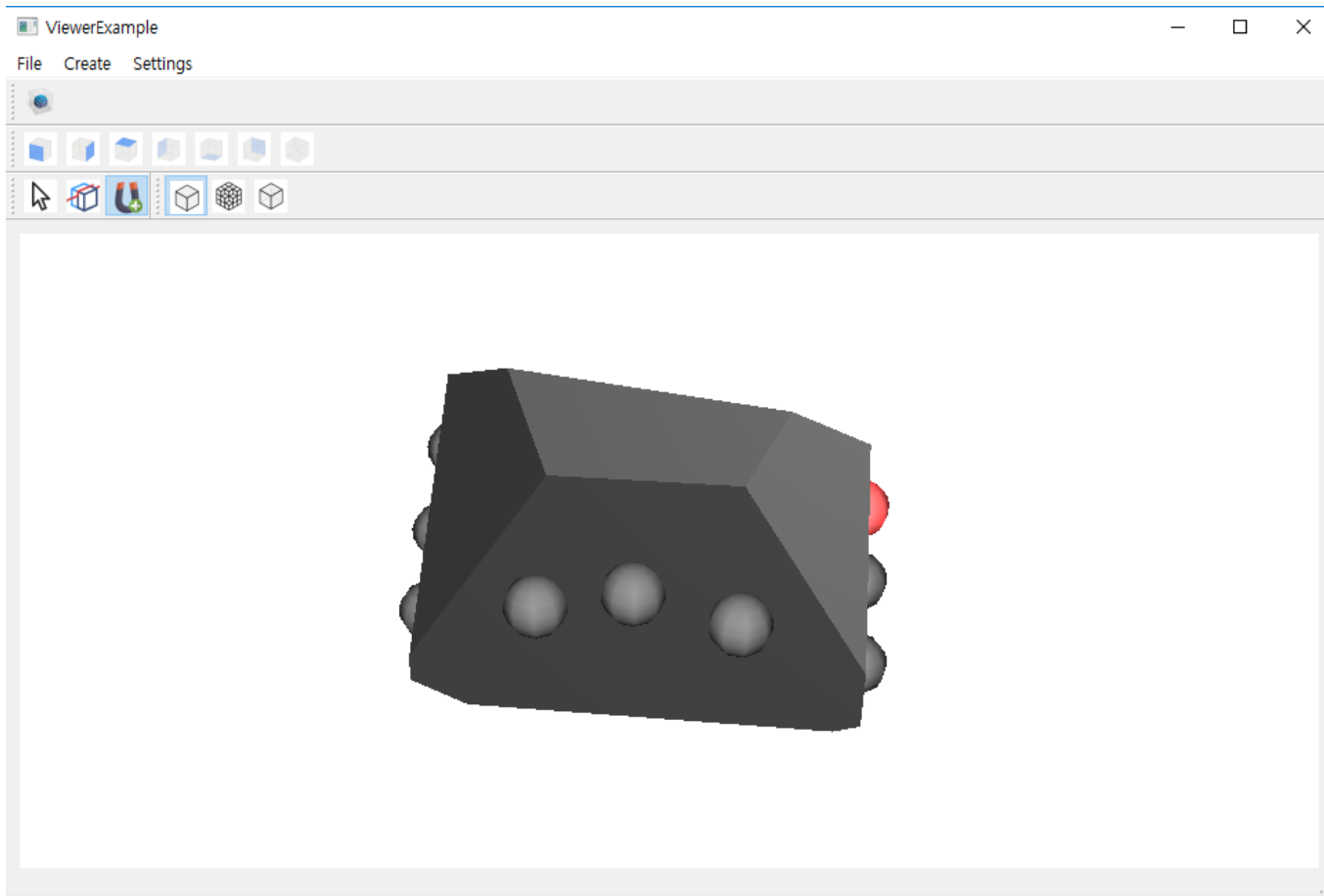
Fig. 41. Control points on the outer surface of Fig. 40 (b)
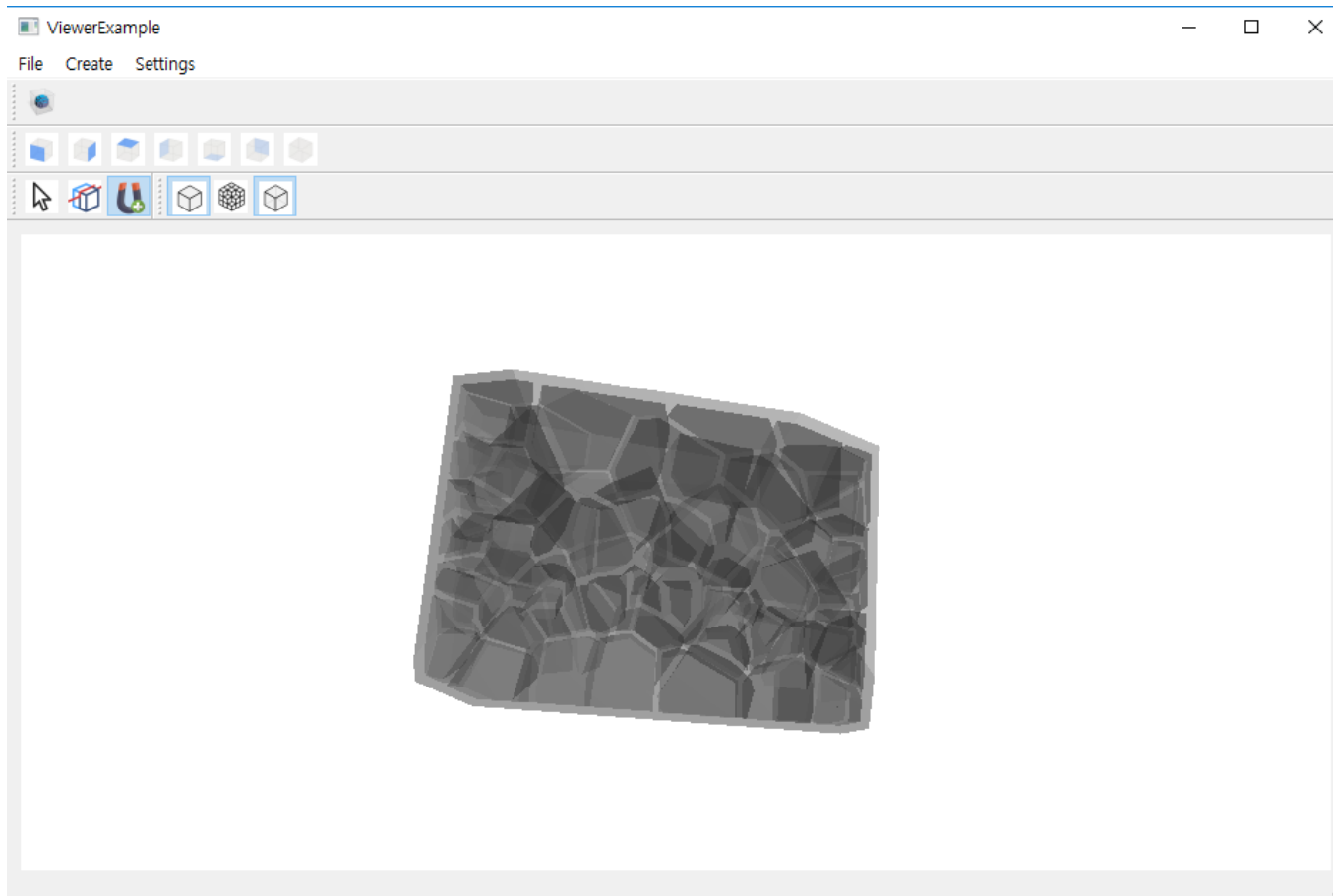
Fig. 42. Inner structure visualization of Fig. 41
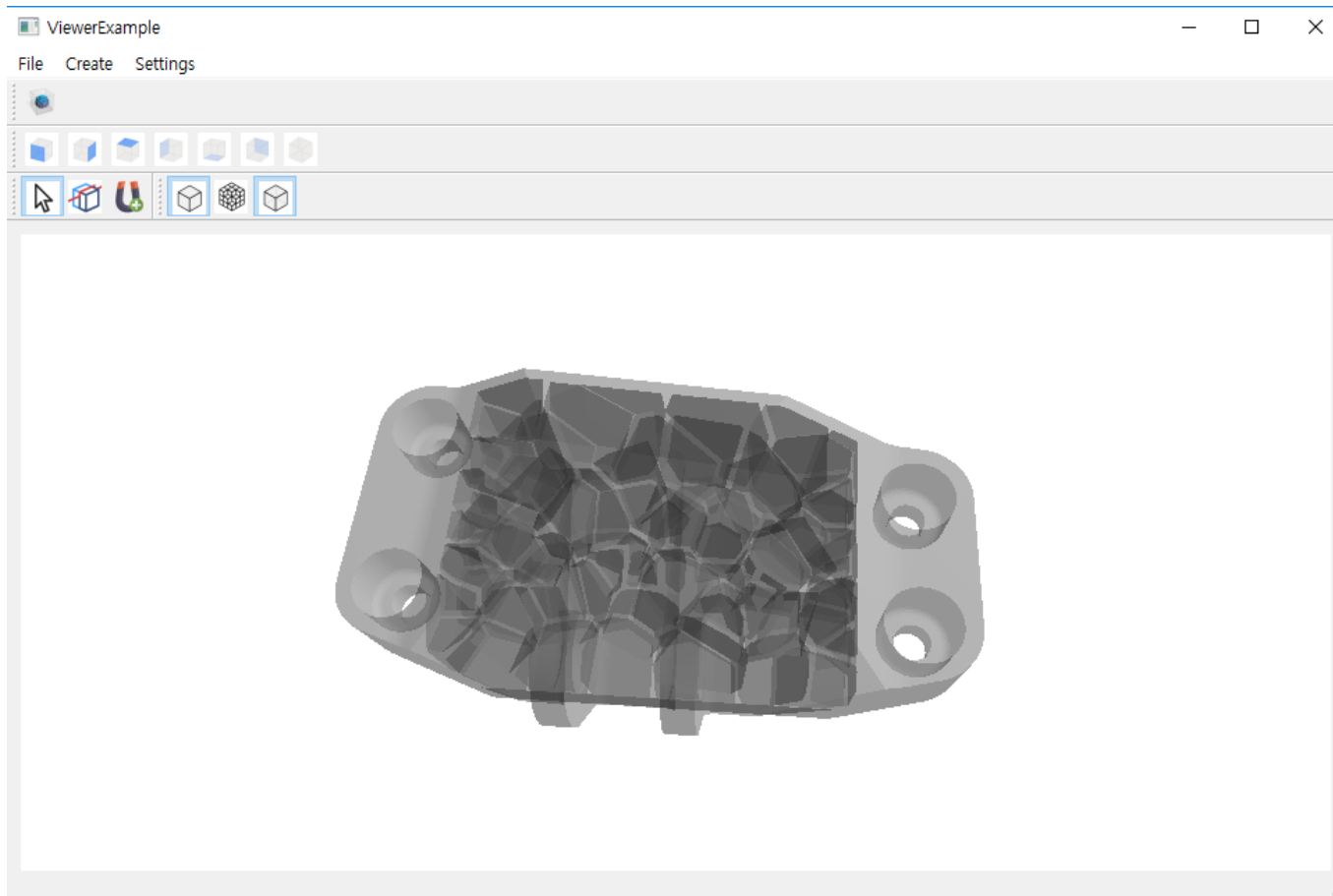
Fig. 43. Visualization after changing outer shell of Fig. 42
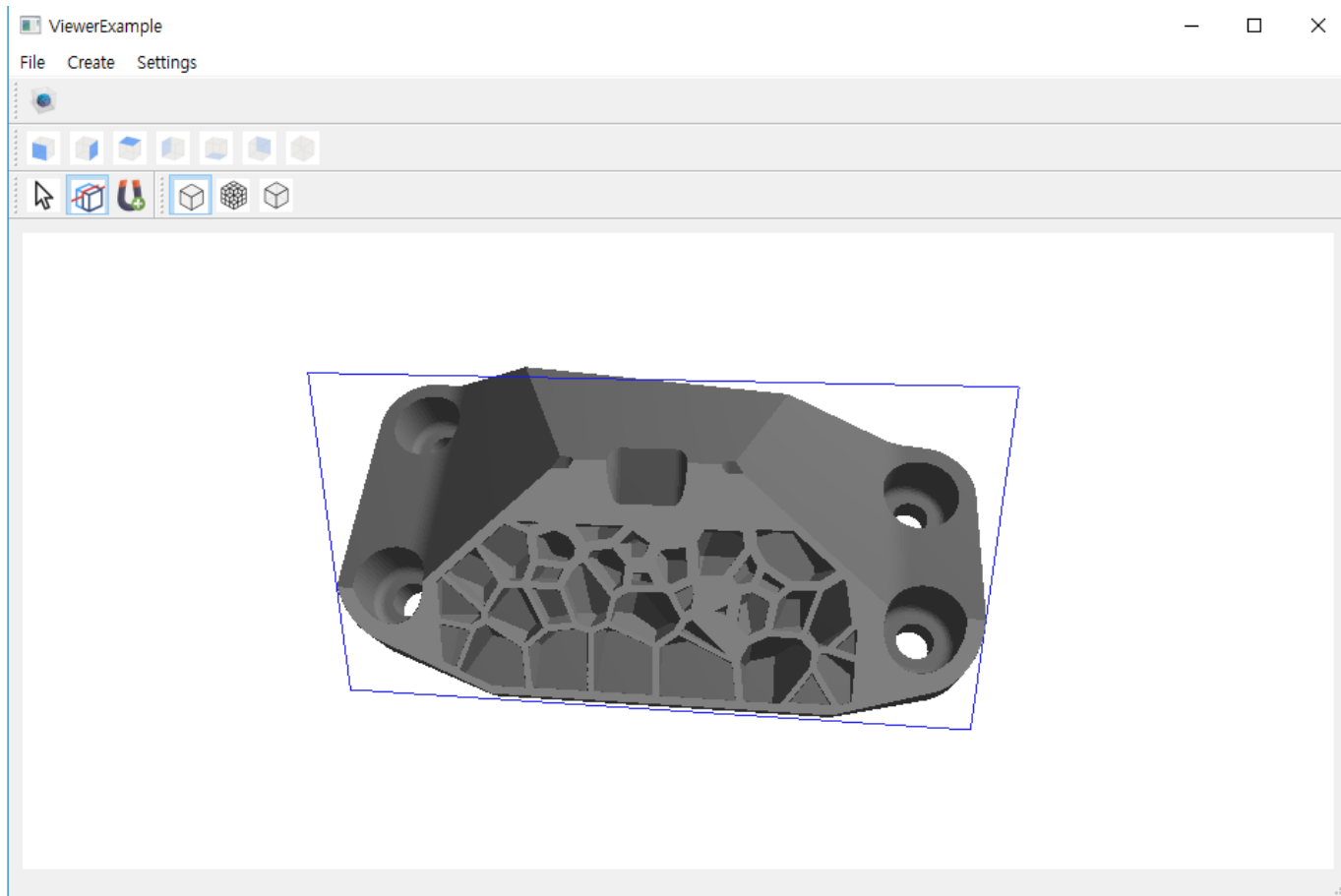
Fig. 44. Clipped visualization of Fig. 43
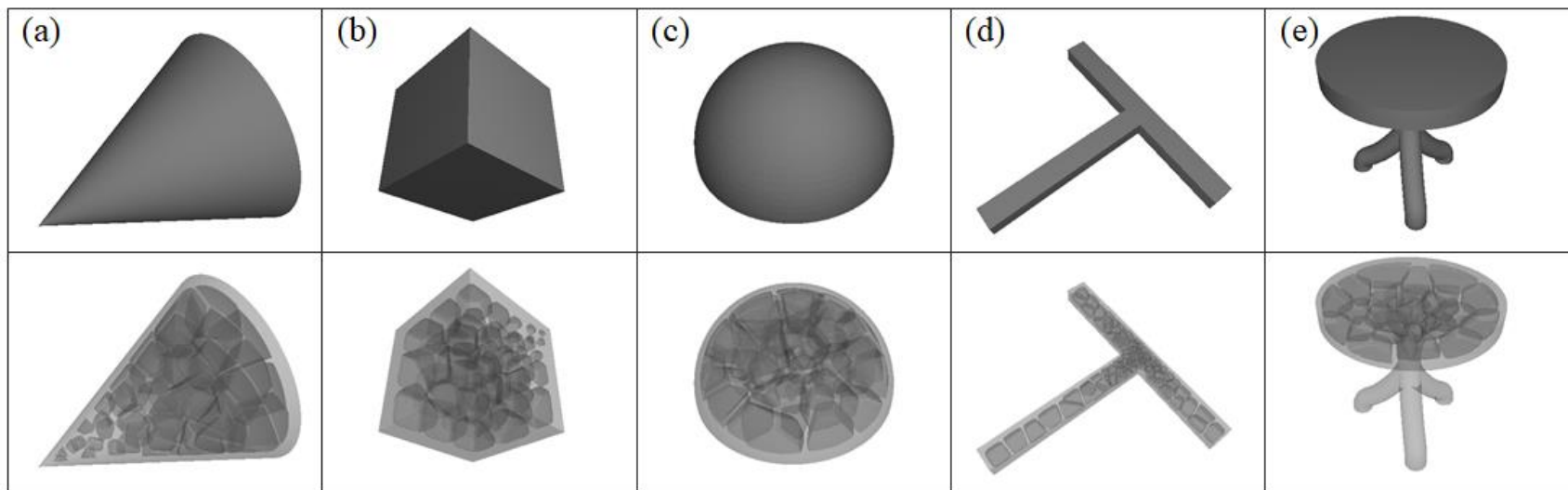
Fig. 45. Tested models. (a) cone, (b) cube, (c) half sphere, (d) T−shape, (e) table

Example	# Mesh vertices of input surface	# Voxels	# Voronoi cells	Time (s)			
				Voxelization	Stippling	Subdivision	Total
Cone	73,186	90 x 90 x 101	50	1.141	0.212	11.097	12.450
Cube	2,402	101 x 101 x 101	51	0.684	1.521	0.142	2.347
Half sphere	2,601	156 x 156 x 79	73	1.065	5.602	0.697	7.364
T-shape	549,936	201 x 16 x 171	149	0.425	0.645	0.487	1.557
Table	12,364	201 x 201 x 201	49	5.151	7.161	0.436	12.748

Table 2. Statistics on the examples

As shown in Table 2, it takes a few seconds to create an inner structure, and the user can fully interact with the application. However, in some cases, certain operations take a little longer than others. In the case of voxelization operation and stippling operation, it can be seen that the time increases in proportion to the voxel resolution.

In the cone case, the subdivision process takes a considerable amount of time. In the subdivision process, the number of vertices in the resulting mesh is increased exponentially in proportion to the number of vertices in the initial mesh. In the current application, each cell is managed as a separate mesh, and Voronoi clipping process increases the number of vertices in each cell near the boundary as shown in Fig. 46. Consequently, applying the subdivision process to those cells increased a considerable amount of time.

On the other hand, in the case of a cube or half sphere, it is believed that the number of vertices in each cell dose not increase excessively during clipping process because the surface mesh is small. In the case of the T-shape, it is believed that the number of cell vertices does not increase excessively because most of the clipping processing is performed with points that exist on the same plane.
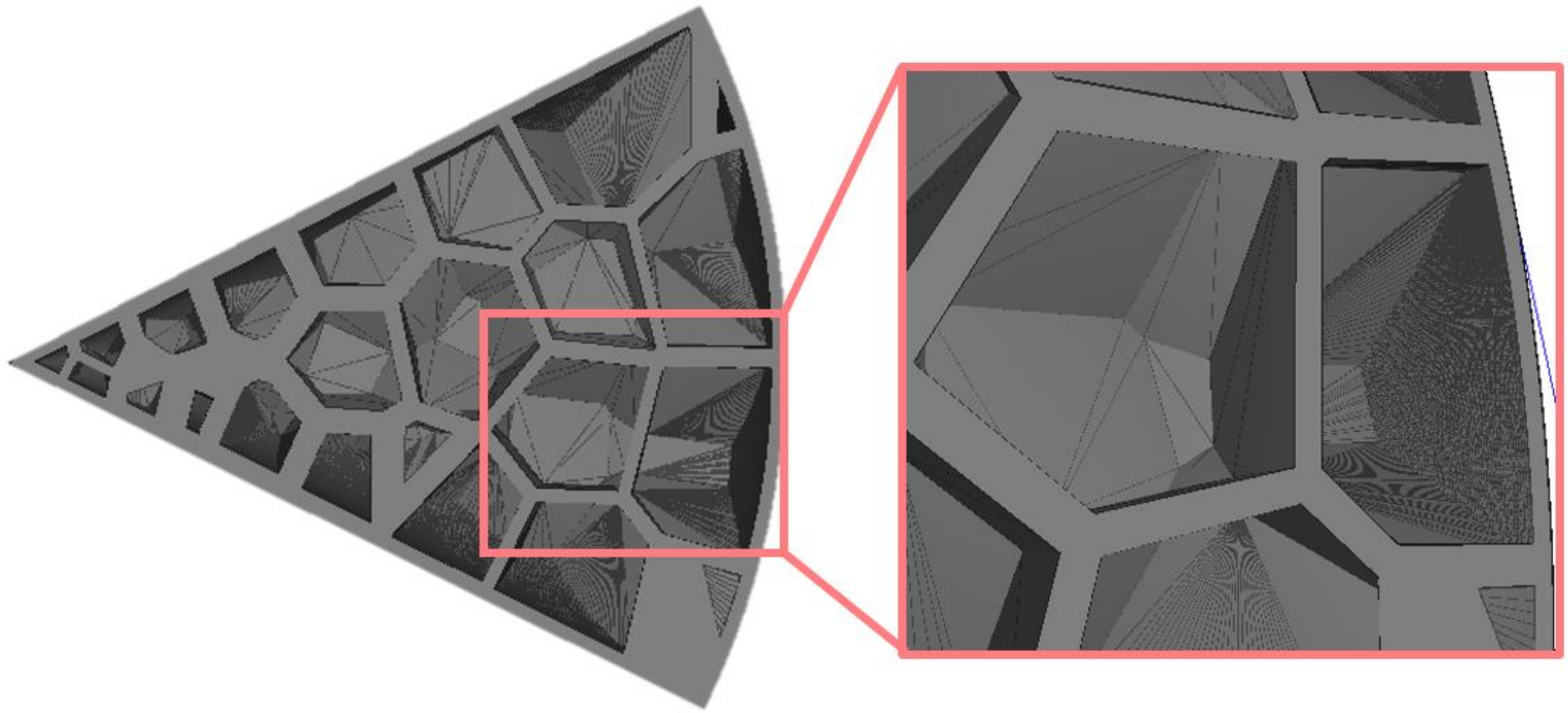
Fig. 46. Many vertices in boundary cells before subdivision process

4.2. Application to strengthen the weak part

As mentioned in Chapter 3, an application that strengthens the weak region can be considered. In this case, only the density of the part to be strengthened increases, and the density of the other regions is relatively low. This means the user can leverage the mechanical properties of the part over its weight. Consequently, the productivity improvement can be achieved. Fig. 47 shows the inner structure of T-shaped parts designed and fabricated through this research.

Fig. 47. T-shaped part with designed inner structure

4.3. Application to control the center of mass

In this study, the carving volume is evenly divided over the entire cell. This means that the weight near the control points is heavy. Therefore, the result of this study can be used in the application that controls the center of gravity as the user's intention.

Xie [56] researched the design of internal structures that did not need support structures during the additive manufacturing process. Similarly, for the solid model such as in Fig. 48 (a), the internal structure was designed to enable self－standing as shown in Fig. 48 (b) by using the program developed in this study. It can also be used to generate the intended eccentricity, as shown in Fig. 49.
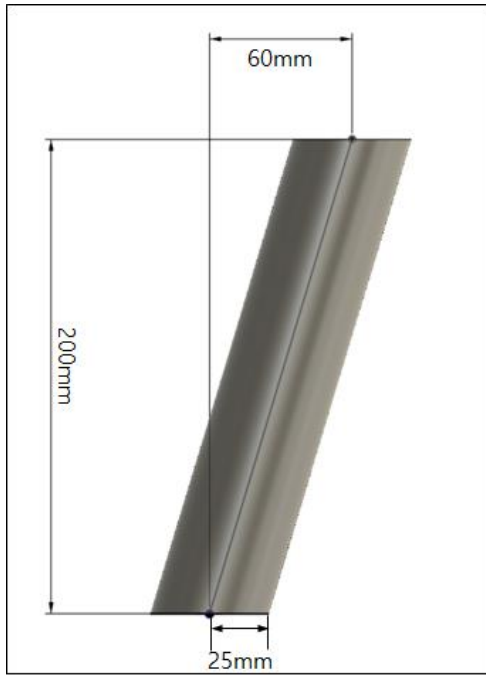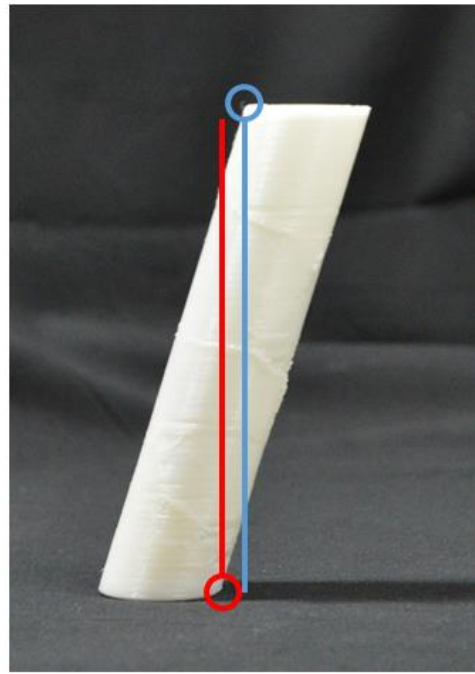
Fig. 48. An example of the object that controls center of mass, (a) the dimension of the object, (b) the object showing self-standing
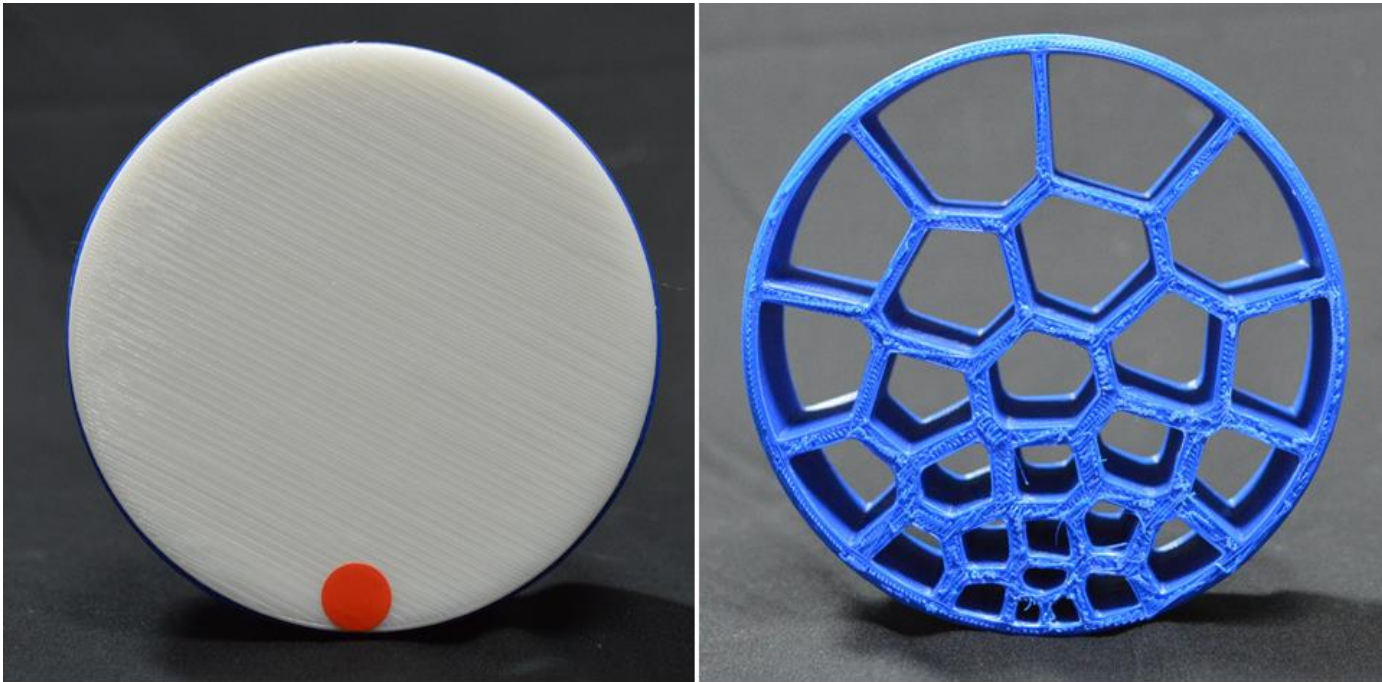
Fig. 49. The intended eccentric structure design, (a) the object with cover showing eccentricity,

(b) the inner structure of the object

Chapter 5. Conclusion

In this study, a new design tool for internal structures of solid models is proposed. This tool is characterized by mimicking the shape that exists in nature, especially the behavior of magnets, to emphasize the user's intuition.

Of course, some additional considerations are required to use this study in practical applications. In this study, the local density can be controlled through several parameters. However, from an analytical point of view, further studies on the relation between mechanical properties and the density of the structures should be performed.

Also, additional user-friendly interfaces are required. The proposed design tool can only control the overall structure. Therefore, it is necessary to implement an interface that allows users to control individual cells. It is also necessary to integrate with interfaces used in the existing CAD system. In this study, the point type interface is suggested as a design tool, but parametric curve or surface type interfaces can be proposed. In this case, it may be possible to perform an internal shape design that is more closely related to the external shape.

In addition, further research on the field equation is needed. Approaching from a probabilistic point of view would be one way. Similar to Alexa et al.'s [57] work of sampling points on a 3D surface

using a probability distribution function, the positions of cells in the internal structure can be obtained through statistical functions. Also, if a tensor field is used rather than a scalar field, the cell anisotropy can be utilized.

However, this study is meaningful in that it presents the inner structure's design tool of solid models that even nonexperts can use quickly and easily. In particular, the development in the field of additive manufacturing is expected to be facilitated by the proposed design tool.

References

[1]   J. P. Groen, J. Wu, and O. Sigmund, "Homogenization-based stiffness optimization and projection of 2D coated structures with orthotropic infill," *Computer Methods in Applied Mechanics and Engineering,* vol. 349, pp. 722-742, 2019.

[2]   J. Wu, N. Aage, R. Westermann, and O. Sigmund, "Infill optimization for additive manufacturing—approaching bone-like porous structures," *IEEE transactions on visualization and computer graphics,* vol. 24, no. 2, pp. 1127-1140, 2017.

[3]   A. P. Fitzpatrick, M. I. Mohanned, P. K. Collins, and I. Gibson, "Design of a patient specific, 3D printed arm cast," *KnE Engineering,* vol. 2, no. 2, pp. 135-142, 2017.

[4]   H. Lin, L. Shi, and D. Wang, "A rapid and intelligent designing technique for patient-specific and 3D-printed orthopedic cast," *3D printing in medicine,* vol. 2, no. 1, pp. 1-10, 2016.

[5]   J. Panetta, Q. Zhou, L. Malomo, N. Pietroni, P. Cignoni, and D. Zorin, "Elastic textures for additive fabrication," *ACM Transactions on Graphics (TOG),* vol. 34, no. 4, p. 135, 2015.

[6]   C. Schumacher, B. Bickel, J. Rys, S. Marschner, C. Daraio, and M. Gross, "Microstructures to control elasticity in 3D printing," *ACM Transactions on Graphics (TOG),* vol. 34, no. 4, p. 136,

2015.

[7]   W. Regli, J. Rossignac, V. Shapiro, and V. Srinivasan, "The new frontiers in computational modeling of material structures," *Computer-Aided Design,* vol. 77, pp. 73-85, 2016.

[8]   W. Gao *et al.,* "The status, challenges, and future of additive manufacturing in engineering," *Computer-Aided Design,* vol. 69, pp. 65-89, 2015.

[9]   B. JENKINS, "3D Engineering: How 3D Direct Modeling Empowers Conceptual Engineering and Enables Simulation-Driven Product Development," *White Paper, Ora Research LLC,* 2009.

[10]   M. Philippa and B. V. Michael, "An ontology of computational tools for design activities," *Proceedings of DRS,* 2018.

[11]   S. Tornincasa and F. Di Monaco, "The future and the evolution of CAD," in *Proceedings of the 14th international research/expert conference: trends in the development of machinery and associated technology*, 2010, vol. 1, no. 1, pp. 11-18.

[12]   A. Azman, F. Vignat, and F. Villeneuve, "Evaluating current CAD tools performances in the context of design for additive manufacturing," in *Joint Conference on Mechanical, Design Engineering & Advanced Manufacturing*, 2014, pp. 1-7.

[13]   M. P. Bendsoe and O. Sigmund, *Topology optimization: theory, methods, and applications*. Springer Science & Business Media, 2013.

[14]   R. Arora *et al.*, "Volumetric Michell trusses for parametric design & fabrication," in *Proceedings of the ACM Symposium on Computational Fabrication*, 2019, p. 6: ACM.

[15]   I. G. Jang and I. Y. Kim, "Computational study of Wolff's law with trabecular architecture in the human proximal femur using topology optimization," *Journal of Biomechanics,* vol. 41, no. 11, pp. 2353−2361, 2008.

[16]   J. Wu, C. Dick, and R. Westermann, "A system for high−resolution topology optimization," *IEEE transactions on visualization and computer graphics,* vol. 22, no. 3, pp. 1195−1208, 2015.

[17]   S. Larsen, O. Sigmund, and J. Groen, "Optimal truss and frame design from projected homogenization−based topology optimization," *Structural and Multidisciplinary Optimization,* vol. 57, no. 4, pp. 1461−1474, 2018.

[18]   A. G. M. Michell, "LVIII. The limits of economy of material in frame−structures," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science,* vol. 8, no. 47, pp. 589−597, 1904.

[19]   S. Daynes, S. Feih, W. F. Lu, and J. Wei, "Design concepts for generating optimised lattice structures aligned with strain trajectories," *Computer Methods in Applied Mechanics and Engineering,* vol. 354, pp. 689–705, 2019.

[20]   S. Daynes, S. Feih, W. F. Lu, and J. Wei, "Optimisation of functionally graded lattice structures using isostatic lines," *Materials & Design,* vol. 127, pp. 215–223, 2017.

[21]   A. Gupta, K. Kurzeja, J. Rossignac, G. Allen, P. S. Kumar, and S. Musuvathy, "Designing and processing parametric models of steady lattices," 2018.

[22]   W. Wang, B. Jüttler, D. Zheng, and Y. Liu, "Computation of rotation minimizing frames," *ACM Transactions on Graphics (TOG),* vol. 27, no. 1, p. 2, 2008.

[23]   J. Rossignac and Á. Vinacua, "Steady affine motions and morphs," *ACM Transactions on Graphics (TOG),* vol. 30, no. 5, pp. 1–16, 2011.

[24]   J. Brennan–Craddock, "The investigation of a method to generate conformal lattice structures for additive manufacturing," © James Brennan–Craddock, 2011.

[25]   Y. Zhou, H. Lu, Q. Ren, and Y. Li, "Generation of a tree–like support structure for fused deposition modelling based on the L–system and an octree," *Graphical Models,* vol. 101, pp. 8–

16, 2019.

[26]   P. Prusinkiewicz and A. Lindenmayer, *The algorithmic beauty of plants*. Springer Science & Business Media, 2012.

[27]   L. Xu and D. Mould, "Procedural tree modeling with guiding vectors," in *Computer Graphics Forum*, 2015, vol. 34, no. 7, pp. 47−56: Wiley Online Library.

[28]   S. Longay, A. Runions, F. Boudon, and P. Prusinkiewicz, "TreeSketch: Interactive Procedural Modeling of Trees on a Tablet," in *SBM*, 2012, pp. 107−120.

[29]   Y. I. Parish and P. Müller, "Procedural modeling of cities," in *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, 2001, pp. 301−308.

[30]   L. Lu *et al.*, "Build−to−last: strength to weight 3D printed objects," *ACM Transactions on Graphics (TOG),* vol. 33, no. 4, pp. 1−10, 2014.

[31]   A. O. Aremu *et al.*, "A voxel−based method of constructing and skinning conformal and functionally graded lattice structures suitable for additive manufacturing," *Additive Manufacturing,* vol. 13, pp. 1−13, 2017.

[32]   O. Fryazinov, T. Vilbrandt, and A. Pasko, "Multi−scale space−variant FRep cellular structures," *Computer−Aided Design,* vol. 45, no. 1, pp. 26−34, 2013.

[33] S. Xu, J. Shen, S. Zhou, X. Huang, and Y. M. Xie, "Design of lattice structures with controlled anisotropy," *Materials & Design,* vol. 93, pp. 443−447, 2016.

[34] J. Berger, H. Wadley, and R. McMeeking, "Mechanical metamaterials at the theoretical limit of isotropic elastic stiffness," *Nature,* vol. 543, no. 7646, pp. 533−537, 2017.

[35] B. Zhang *et al.*, "The biomimetic design and 3D printing of customized mechanical properties porous Ti6Al4V scaffold for load−bearing bone reconstruction," *Materials & Design,* vol. 152, pp. 30−39, 2018.

[36] L. Wang, J. Kang, C. Sun, D. Li, Y. Cao, and Z. Jin, "Mapping porous microstructures to yield desired mechanical properties for application in 3D printed bone scaffolds and orthopaedic implants," *Materials & Design,* vol. 133, pp. 62−68, 2017.

[37] M. P. Bendsoe and N. Kikuchi, "Generating optimal topologies in structural design using a homogenization method," 1988.

[38] H. Fredricson, T. Johansen, A. Klarbring, and J. Petersson, "Topology optimization of frame structures with flexible joints," *Structural and multidisciplinary optimization,* vol. 25, no. 3, pp. 199−214, 2003.

[39] M.−J. Kim, G.−W. Jang, and Y. Y. Kim, "Application of a ground beam−joint topology optimization method for multi−piece

frame structure design," *Journal of mechanical Design,* vol. 130, no. 8, 2008.

[40]   T. J. Goodman and R. Spence, "The effect of computer system response time variability on interactive graphical problem solving," *IEEE Transactions on Systems, Man, and Cybernetics,* vol. 11, no. 3, pp. 207−216, 1981.

[41]   T. Zheng, J. Wei, Z. Shi, T. Li, and Z. Wu, "An Overview of Modeling Yarn'ｓ 3D Geometric Configuration," *Journal of Textile Science and Technology,* vol. 1, no. 01, p. 12, 2015.

[42]   L. Piegl and W. Tiller, *The NURBS book*. Springer Science & Business Media, 1996.

[43]   S.−H. Jang, S.−H. Na, and Y.−L. Park, "Magnetically assisted bilayer composites for soft bending actuators," *Materials,* vol. 10, no. 6, p. 646, 2017.

[44]   M. Wang, L. He, and Y. Yin, "Magnetic field guided colloidal assembly," *Materials Today,* vol. 16, no. 4, pp. 110−116, 2013.

[45]   C. Jakob and H. Konietzky, "Particle Methods," *An Overview Freiberg,* vol. 24, 2012.

[46]   J. L. Domingos, F. M. Peeters, and W. Ferreira, "Self−assembly of rigid magnetic rods consisting of single dipolar beads in two dimensions," *Physical Review E,* vol. 96, no. 1, p. 012603, 2017.

[47] G. Akinci, M. Ihmsen, N. Akinci, and M. Teschner, "Parallel Surface Reconstruction for Particle-Based Fluids," *Computer Graphics Forum,* vol. 31, no. 6, pp. 1797-1809, 2012.

[48] J. Görtler, M. Spicker, C. Schulz, D. Weiskopf, and O. Deussen, "Stippling of 2D scalar fields," *IEEE transactions on visualization and computer graphics,* vol. 25, no. 6, pp. 2193-2204, 2019.

[49] O. Deussen, M. Spicker, and Q. Zheng, "Weighted linde-buzo-gray stippling," *ACM Transactions on Graphics (TOG),* vol. 36, no. 6, pp. 1-12, 2017.

[50] D. F. Watson, "Computing the n-dimensional Delaunay tessellation with application to Voronoi polytopes," *The computer journal,* vol. 24, no. 2, pp. 167-172, 1981.

[51] C. Rycroft, "Voro++: A three-dimensional Voronoi cell library in C++," Lawrence Berkeley National Lab.(LBNL), Berkeley, CA (United States)2009.

[52] S. Lloyd, "Least squares quantization in PCM," *IEEE transactions on information theory,* vol. 28, no. 2, pp. 129-137, 1982.

[53] W. Zhuo and J. Rossignac, "Curvature-based offset distance: Implementations and applications," *Computers & Graphics,* vol. 36, no. 5, pp. 445-454, 2012.

[54] E. Catmull and J. Clark, "Recursively generated B-spline surfaces on arbitrary topological meshes," *Computer-aided design,* vol. 10, no. 6, pp. 350-355, 1978.

[55] D.-M. Yan, W. Wang, B. Lévy, and Y. Liu, "Efficient computation of clipped Voronoi diagram for mesh generation," *Computer-Aided Design,* vol. 45, no. 4, pp. 843-852, 2013.

[56] Y. Xie and X. Chen, "Support-free interior carving for 3D printing," *Visual Informatics,* vol. 1, no. 1, pp. 9-15, 2017.

[57] M. Alexa, S. Rusinkiewicz, D. Nehab, and P. Shilane, "Stratified point sampling of 3D models," in *Proc. Eurographics Symp. on Point-Based Graphics*, 2004, pp. 49-56.

Abstract (Korean)

솔리드 모델의 내부 구조를 위한 직관적인 자석 모사 디자인 도구

감 동 욱

서울대학교 대학원

기계항공공학부

본 연구에서는 주어진 외부 형상에 대해, 사용자가 상호작용하며 내부 구조를 모델링할 수 있는 새로운 디자인 도구를 제시하였다. 이 디자인 도구는 자연에서 자석과 같은 끌개가 동작하는 방식을 모사한다. 본 연구에서는 사용자와의 빠른 상호작용 및 직관과 유사한 형태를 만들기 위해, 보로노이 구조를 그 기반으로 두고있다. 디자인 도구와 생성되는 보로노이 구조 간의 관계를 기술함으로써, 사용자가 상호작용할 수 있는 파라미터들을 제시하였다. 사용자는 디자인 도구의 위치, 개수 및 각 도구가 가지는 파라미터의 조절을 통해 내부 구조를 생성, 변경할 수 있다. 각 보로노이 셀의 스케일 조절을 통해 실제 제품 제작에 활용될 수 있도록 하였으며, 그 내부 구조가 부드럽게 연결될 수 있도록 Subdivision 알고리즘을 응용하였다. 이러한 디자인 도구의 활용처로, 제품 무게 중심의 조절 및 부품이 가지는 특정 부분의 역학적 성질을 좋게 만드는 두 가지 응용분야를 제시하였다. 결론적으로 적층 제조 방법으로 인해 변화

하는 제조업의 패러다임을 선도할 수 있도록, 내부 구조를 설계할 수 있

는 디자인 도구라는 새로운 개념을 제시하였다.

주요어 : CAD 도구, 내부 구조 디자인, 보로노이 구조, 자석 모방

학 번 : 2014-21864