

RESEARCH

Open Access



Automatic Segmentation and 3D Printing of A-shaped Manikins using a Bounding Box and Body-feature Points

Jin Young Jung^{1†}, Seonkoo Chee^{2†} and In Hwan Sul^{3*} 

*Correspondence:

snowman0@kumoh.ac.kr

[†]Jin Young Jung and Seonkoo Chee contributed equally to this work

³ Associate Professor, Dept. of Materials Design Engineering, Kumoh National Institute of Technology, Gumi 39177, Republic of Korea

Full list of author information is available at the end of the article

Abstract

A novel algorithm for 3D-printing technology was proposed to generate large-scale objects, especially A-shaped manikins or 3D human body scan data. Most of the conventional 3D printers have a finite printing volume, and it is the users' work to convert the target object into a printable size. In this study, an automatic three-step segmentation strategy was applied to the raw manikin mesh data until the final pieces had a smaller size than the 3D printer's maximum printing volume, which is generally called "beam length". Human body feature point information was adopted for fashion and textile researchers to easily specify the desired cutting positions. A simple bounding box, especially orienting bounding box, and modified Boolean operator were proposed to extract the specified segments with computational stability. The proposed method was applied to graphically synthesized manikin data, and 1/8, 1/4, and 1/2 scale manikins were successfully printed, minimizing the amount of support structure.

Keywords: 3D printing, Automatic segmentation, Manikin, Bounding box, Body-feature point

Introduction

Three-dimensional (3D) printing or additive manufacturing (ISO, 2015) is definitely one of this era's hottest keywords. It was initially developed to rapidly generate prototypes of the final product in mechanical engineering and gradually found application in different industries (Ngo et al. 2018). As the initial patents for fused decomposition modeling (Crump, 1992) and selective laser sintering (Deckard, 1989) expired, studies on 3D printing accelerated, especially concerning filament material and printer-control technology. In this context, material scientists and engineers are heavily interested in the former topic. Therefore, composite structures (Wang et al. 2017) have also been studied for fabricating better filaments, in addition to single materials such as plastics (Talataisong et al. 2018), metals (Chen et al. 2019), cement (Perrot et al. 2016), and biomedical tissues (Murphy & Atala, 2014).

By the way, the latter problem becomes more burden to the 3D printer users. Printer control includes preparing 3D data, converting the data into 3D printer-friendly format, and post-processing the final product. First, 3D data can be prepared by graphically

synthesizing, or more preferably by reverse engineering. If the target object exists, the user can acquire mesh data by simple 3D scanning. Three-dimensional scanning technology has been widely spread, and even some smartphones support this feature. On the contrary, there are some cases when there is no prototype to scan, and the user has to synthesize the data. Fortunately, there exist many tools to generate 3D mesh data, from professional computer-aided design (CAD) or modeling software such as Maya[®], MAX[®], Rhinoceros 3D[®], and AutoCAD[®], to user-friendly software like SketchUp[®] and TinkerCAD[®]. Especially in the apparel industry, 3D human body scanners have been used to acquire human body shape data for several decades. If the manikin data is small enough to fit the 3D printer, printing manikin is easy. The user only has to convert the 3D data into g-code, print the g-code, and remove redundant support structure. Sometimes mechanical or chemical polishing is added to make a smooth surface.

However, manikins, such as garment display or car crash safety dummies, are generally bigger than printers. In this case, the user should have an additional skill to modify the raw manikin data into smaller pieces. The conventional slicing software supports a limited segmentation feature, which operates for simple shapes only. Otherwise, the user can use programming language or tools such as C, C++, Python[®], or Matlab[®] to freely manipulate the 3D data. However, it is still a burden to dissect a complex-shaped large object like a manikin to a user-defined size with these tools.

Therefore, we attempted to develop a 3D-printing technique optimized for human manikins in this study. Especially, the new method should satisfy the following conditions.

- [condition #1] degree of freedom: The final manikin generally has a whole body shape, but sometimes partial segments like the bodice may be needed. It should be able for the user to specify desired body parts from the raw manikin data.
- [condition #2] ease of implementation: The method should be numerically simple and easy to implement in any programming language (C/C++/Python) or tools (Matlab).
- [condition #3] aesthetics: It is preferred to make the final printed parts symmetric for visual quality (Luo et al. 2012).
- [condition #4] printer size constraint: The method should consider the maximum printing volume.
- [condition #5] minimization of the support structure: The method should minimize the support structure to reduce filament wastage and surface ruggedness.
- [condition #6] numerical efficiency: The calculation speed should be as fast as possible.
- [condition #7] geometric stability: The method should generate a geometrically valid (manifold) model, which has a closed-volume.

This paper combined several multi-disciplinary techniques. First of all, feature points were used to satisfy condition #1. Feature points and bone/joint information were automatically generated from the manikin shape based on our previous method (Sul & Kang, 2010) in advance. Also, a simple 3D box, also known as a bounding box, was used to meet conditions #2–#7. We found that applying the conventional Boolean operation to

a B-rep mesh data yielded a non-manifold (not-closed-volume) model, leading to 3D printing failure. Therefore a new modified Boolean operation was devised to meet conditions #6 and #7. “[Literature review](#)” section summarizes related previous researches, with a brief introduction to our previous work. “[Experimental](#)” and “[Modeling](#)” section shows the experimental and algorithmic details. Our method was applied to the 3D printing of manikins with three different scales. The results are shown and discussed in “[Results and discussion](#)” section.

Literature review

Automatic segmentation of a large-scale object for 3D printing

Additive manufacturing of large-scale prototypes is needed in various areas, such as 3D-printed houses for fast construction (Bos et al. 2016), dummies for car-safety testing (Boström et al. 2000), human organs for biomedical applications (Thielen & Delbressine, 2016), and manikins for the apparel industry (Nayak & Padhye, 2017). Because most conventional 3D printers do not support large-scale printing features, the user should decompose raw mesh data into smaller parts using 3D CAD or modeling tools. Several researchers already tackled this printer control issue; MIT’s “Chopper” algorithm was the pioneer in slicing large objects (Luo et al. 2012). It decomposes a big model into 3D-printable sizes considering printing volume (called as “beam length” in this article), connector feasibility, finite element method-based structural soundness, and even esthetics such as user guidance or model symmetry (Luo et al. 2012) as objective functions. Several other approaches followed in the footsteps of this algorithm, including “Q-CHOPPER” (Nicholson, 2016), “Dapper” (Chen et al. 2015), and “PackMerger” (Vanek et al. 2014).

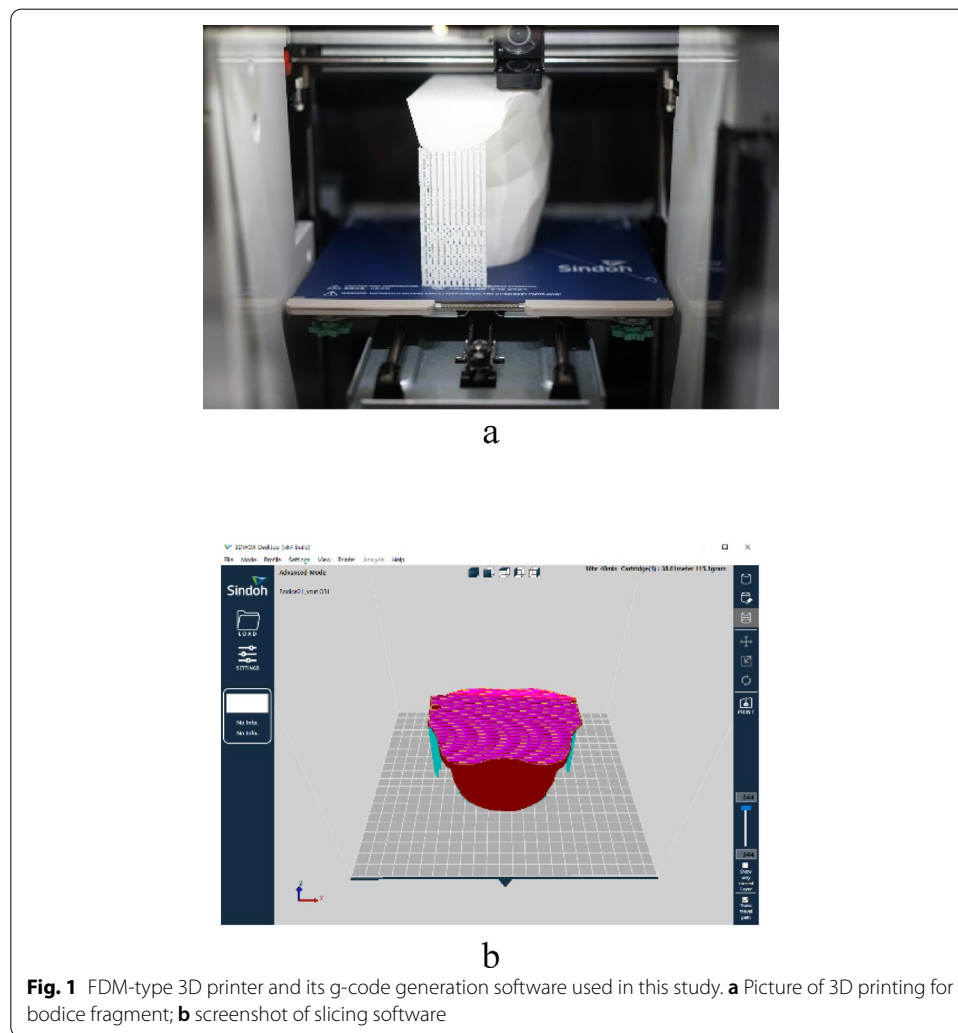
Although these slicing algorithms are versatile and highly efficient, they also have several cons, such as calculation speed and complex data structure. For example, the “Chopper” algorithm partitions a target object with a cutting plane of infinite size and generates a binary space partition (BSP) structure for each trial. The problem is that it seeks a local optimum with no prior knowledge of object geometry. Therefore, the algorithm starts segmentation from the bottom layer. Thus, the two parts acquired by slicing are again partitioned using another cut plane until the part size is smaller than the beam length. In this way, one BSP tree is finally acquired. This process is repeated, varying the vertical position by 0.5 intervals and the normal vector of the slicing plane in 129 directions. This results in 300,000 candidate BSP trees, depending on the object size (Nicholson, 2016). Finally, the most optimal BSP tree is chosen by applying object functions. Although the problem size is big, previous methods showed a relatively good performance using the parallelization technique. For example, the “Chopper” algorithm requires 2.3 min to decompose “human” data with 4706 vertices into 8 parts (refer to Figure 18). Another disadvantage of the currently available 3D-printing algorithms is related to the slicing plane. A plane with infinite size can cut an object in an undesired shape. Let us assume that the user wants to cut manikin data vertically into upper and lower parts and put the slicing plane around the waist. However, the slicing plane would cut not only the bodice but also the two arms simultaneously.

Minimization of the support structure

3D printing is composed of continuous stacking of polymeric resins layer by layer, similarly to the Hanoi tower puzzle. If the new disk is smaller than the lower disk, there will be no wastage. The opposite case is known as an “overhang” structure, which needs “supports” to maintain the whole structure during solidification. It is also possible to use PVA-based water-soluble support (Duran et al. 2015). In other cases, the support structure leads to waste of filament and undesirable surface artifacts. It is probably the main bottleneck for the industrialization of 3D printing. Many teams have tackled this issue. The simplest method is to find optical printing direction, reducing the total amount of support structure. (Ezair et al. 2015) proposed GPU-based evaluation algorithm which can predict the optical printing direction. The other approach is changing the shape of supports to Y- or tree-shape (Vanek et al. 2014), bridge (Shen et al. 2016), honeycomb (Lu et al. 2014), and unit-cell (Vaidya & Anand, 2016). The support structure can also exist inside the object, especially when it is a hollow one. Medial axis transform (Zhang et al. 2015) and skin-frame structure (Wang et al. 2013) were also adopted to minimize the internal support structure. These works give successful results and can be used in any geometry, especially when the target object’s shape cannot be modified. However, a better way would be to convert the original shape into a support-free one. (Demir et al. 2018) decomposed the object into convex shapes reducing the filament usage to about 65 vol% for some geometry. (Wei et al. 2017) reduced the filament usage to 65~83 vol % by using the skeletal approach. Both methods showed a good result in a reasonable time (15~55 seconds). However, the segmentation is affected by the geometry and cannot be controlled by the user. The human body can have various sizes, and these methods cannot generate consistent results. The other method is to decompose the original object into a group of simple shapes whose upper layer always had a smaller area than its lower one, like the Hanoi-tower. For example, (Wang et al. 2017) have chosen the so-called genus-zero topology, such as cone, pyramid, ellipses, and sphere. However, their method reduced the filament amount to 53~76 vol% but needed considerable time (18~60 minutes). This paper used a similar approach, except that the computation time was much reduced.

Our previous works

The proposed method’s validity was tested in our previous work (Kim & Sul, 2018). Initially, a three-dimensional box of unit size was created in CAD software. Its size, position, and orientation were changed to fit the specific body part. 1/8 scale of human manikin data was loaded, and then the Boolean operator was applied between the box and the manikin data. This process was repeated until a total of 20 segments were acquired. The printing direction was found by rotating each segment and measuring the filament needed from the g-code. All the operation was done manually, and the result showed that the amount of support structure dropped from 63.6 vol% into 1.7 vol%. This research aims to automatize our previous work using simple architecture so that non-professional end-users may easily apply our method use in their application.



Note that feature points are an essential part of our method. They specify the region where the Boolean operation is needed. It does not matter if they are input manually or automatically. Our team devised an automatic feature point detection algorithm in our previous work (Sul & Kang, 2010), and it was used again in this work. It loads the manikin or human body data with an “A”-pose and finds hundreds of cross-sections of the data. By assessing the number of closed curves, feature points could be found.

Experimental

Manikin data

A graphically synthesized body was used to minimize possible numerical error. The body data (“Masha” at Turboquid.com) was bought online in Wavefront.OBJ format. It contained 6,960 vertices and 13,672 triangular mesh elements, with a vertical size of 171.3 cm. The initial pose of the manikin was assumed to be A-shaped to facilitate the following feature-point-extraction process. Throughout this study, all 3D data (including raw manikin mesh) was displayed using our homemade MS-Windows-based program, using C++ language with the Embacadero C++ Builder XE5 compiler.

3D printer

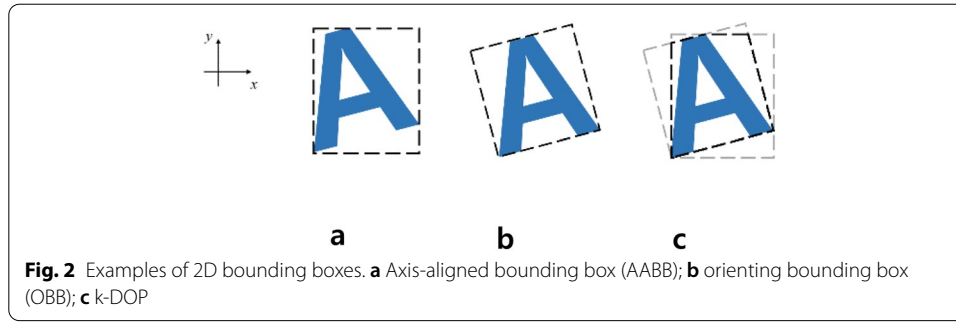
Fused-deposition modeling (FDM)-type printer (Sindoh DP-202, Korea) was used with poly(lactic acid) (PLA) filaments in this study (Fig. 1a). The printer's bundled software (3DWOX) was used to generate the g-code (Fig. 1b) by keeping the default options unchanged except for the build-plate generation option being turned off and the support structure being turned on to a zig-zag pattern in the necessary positions. The maximum printable size was $200 \times 200 \times 189 \text{ mm}^3$. The printed parts' weight was measured thrice using an electronic balance (HS1100A, Hansung Corporation, Korea; max precision 0.01 g) before and after removing the support structure. The theoretical amount of the support structure was calculated by the bundle-slicing software during g-code generation.

Modeling

Terminology

The following concepts are used throughout this study.

- Decomposing, partitioning, and segmenting: These terms have the same meaning and represent the conversion of a big object into a group of smaller ones (Demir et al. 2018; Jadoon et al. 2018; Nicholson, 2016).
- Boundary-representation (B-rep) and Constructive Solid Geometry (CSG): There are mainly two ways of describing 3D objects in CAD or computer graphics. Most commercial CAD tools, such as AutoCAD®, Rhinoceros 3D®, SketchUp®, and so on, use the B-rep method with which a 3D shape is described with vertices, edges, and faces. It is straightforward and intuitive but may contain minute error coming from floating-point round-off error. The other method, CSG, was devised to generate exact or errorless objects, especially for mechanical engineering. It does not describe the target shape directly. Instead, it uses union, intersection, or difference operations between the given simple shapes such as cubes or boxes. CATIA, SolidWorks, and TinkerCAD® are examples of CSG type CAD tools.
- 6 body-parts and segments: Our algorithm uses 3 steps to decompose the input manikin mesh into smaller pieces. In the first step, the whole-body manikin was decomposed into head, bodice, arms, and legs, which collectively represent “6 body parts”. The second (“V-CUT”) and third (“PIE-CUT”) step segmented the 6 body parts again into smaller sizes so that they could fit the 3D printer. These pieces were designated as “segments”.
- V-CUT and PIE-CUT: The processes by which the 6 body parts are partitioned into segments are designated as V-CUT or PIE-CUT. The former partitions the object in a vertical direction (along with manikin's height), while the latter partitions in the horizontal direction like cake-cutting or pie-cutting (Stromquist, 1980).
- Slicing: This term refers to the generation of g-codes from the cross-section curve of segments in 3D printing (Wang et al. 2015), using software such as Simplify3D®, Ultimaker Cura®, Autodesk Netfabb®, and PrusaSlicer (former Slic3r).
- Beam length: This term refers to the shortest axis length of the printing volume of the 3D printer, originally defined by Luo and co-workers (Luo et al. 2012). The final



3D printed segment's size in its longest axis direction should be smaller than the given beam length.

- **Body feature points:** This term refers to some characteristic points on the human body surface (ISO, 2017). In the study, they also include bones and joint points (Sul & Kang, 2010). Bone points are those with a non-zero degree of freedom, while a joint is the center position of two bone points and does not have skeletal deformation.

Bounding box and eigenvectors

The bounding box is defined as the smallest closed volume that encloses the target shape (Teschner et al. 2005). Among the several types of bounding boxes shown in Fig. 2, the oriented bounding box (OBB; Fig. 2b) was chosen for vertical slicing (V-CUT). Additionally, pie-shaped boxes were used for horizontal slicing (PIE-CUT).

The support for input manikin mesh data with l vertices and m faces can be expressed as shown in Eq. (1), where $\langle \rangle$ means a vector list and \mathbf{p}_i , \mathbf{f}_m , and \mathbf{n}_m are the position vectors of vertices, face information, and face normal vector, respectively.

$$A(\langle \mathbf{p}_l \rangle, \langle \mathbf{f}_m, \mathbf{n}_m \rangle) \quad (1)$$

Solving Eq. (2) and (3) yields eigenvectors (\mathbf{v}_1 , \mathbf{v}_2 , and \mathbf{v}_3) corresponding to mesh A via principal component analysis (PCA) or modal analysis (Kim & Sul, 2018; Pentland & Williams, 1989).

$$\mathbf{p}_i \cdot \mathbf{p}_j = \begin{bmatrix} \mathbf{p}_{i,x} \cdot \mathbf{p}_{j,x} & \mathbf{p}_{i,x} \cdot \mathbf{p}_{j,y} & \mathbf{p}_{i,x} \cdot \mathbf{p}_{j,z} \\ \mathbf{p}_{i,y} \cdot \mathbf{p}_{j,x} & \mathbf{p}_{i,y} \cdot \mathbf{p}_{j,y} & \mathbf{p}_{i,y} \cdot \mathbf{p}_{j,z} \\ \mathbf{p}_{i,z} \cdot \mathbf{p}_{j,x} & \mathbf{p}_{i,z} \cdot \mathbf{p}_{j,y} & \mathbf{p}_{i,z} \cdot \mathbf{p}_{j,z} \end{bmatrix} \quad (2)$$

$$(\mathbf{p}_i \cdot \mathbf{p}_j) \cdot \mathbf{v} = \lambda \mathbf{v} \quad (3)$$

These eigenvectors represent the local axis of each OBB. Meanwhile, the OBB size is controlled by span values along each eigenvector. For example, span values ($\psi_{1,\min}$, $\psi_{1,\max}$) with respect to \mathbf{v}_1 can be calculated as shown in Eqs. (4) and (5). Therefore, each OBB can be described with a triple set, (\mathbf{v}_1 , $\psi_{1,\min}$, $\psi_{1,\max}$), (\mathbf{v}_2 , $\psi_{2,\min}$, $\psi_{2,\max}$), or (\mathbf{v}_3 , $\psi_{3,\min}$, $\psi_{3,\max}$), which needs only 15 real numbers. AABB has the same data structure, except that the three vectors (\mathbf{v}_1 , \mathbf{v}_2 , and \mathbf{v}_3) coincide with the global axes. Figure 3 shows an example of eigenvector evaluation using Mathematica software.

$$\psi_{1,\min} = \min(\langle \mathbf{v}_1 \cdot \mathbf{p}_i \rangle) \quad (4)$$

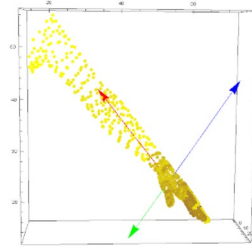


Fig. 3 Example of PCA-based principal axes finding for the left arm using Mathematica® software

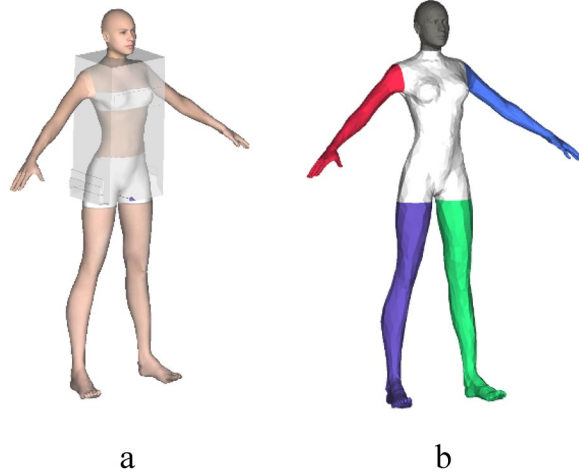


Fig. 4. 6-segment preparation. **a** boolean operation of the raw manikin and temporary bodice bounding volume, **b** extracted bodice segment and the other 5 body segments

$$\psi_{1,\max} = \max (\langle \mathbf{v}_1 \cdot \mathbf{p}_i \rangle) \quad (5)$$

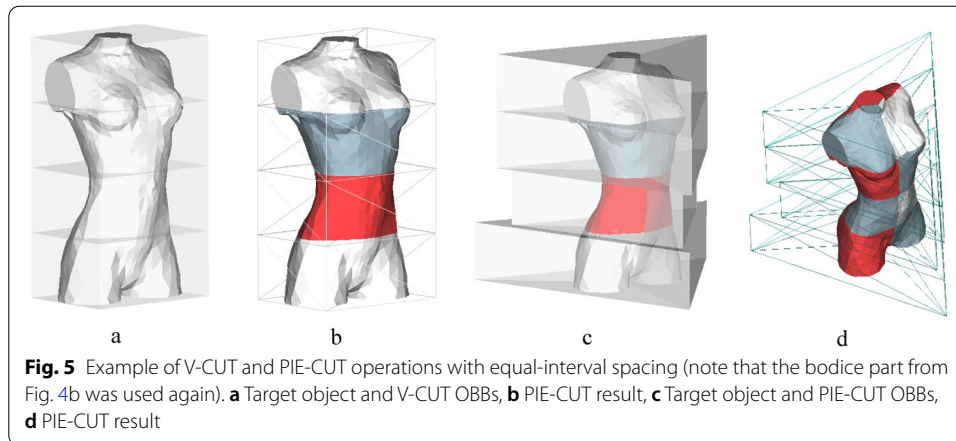
Boolean operation

Our slicing strategy is based on constructive solid geometry (CSG), which results in another shape from two given objects by Boolean operations such as union, intersection, or difference. The intersection operator was mainly used in this study to extract specific body parts from raw manikin data. FinalMesh® Boolean API in dynamic link library format (PelikanSoftwareKft., 2020) was chosen for the Boolean operation engine.

Three-step slicing

Decomposition into 6 body parts

To make the main slicing procedures (V-CUT and PIE-CUT) easier, raw manikin data was decomposed into 6 body parts, using feature-point information. A temporary bounding volume, i.e., AABB of bodice cross-sectional curves, was prepared in advance.



Using Boolean operations between the raw manikin mesh and temporary volume (Fig. 4a), the bodice part, as well as the other 5 parts (head, arms, and legs) were acquired (Fig. 4b). Note that different object colors throughout the paper were intentionally used to show that Boolean operations were successfully conducted.

V-CUT

If the object is relatively small (for example, 1/8 scale manikin), it can be decomposed using only vertical-direction slicing of the 6-parts. The V-CUT operation was realized by preparing several stacked OBBs around each of the 6 parts. The number and size of the OBBs were controlled using the beam length L_b . Herein, we define L_b as the least value among the 3D printer's maximum printable x-, y-, and z-dimensions (Luo et al. 2012). Without further feature-point information, the considered OBBs exhibited an equal vertical size, and the number of OBBs (N_s) was set as shown in Eq. (6),

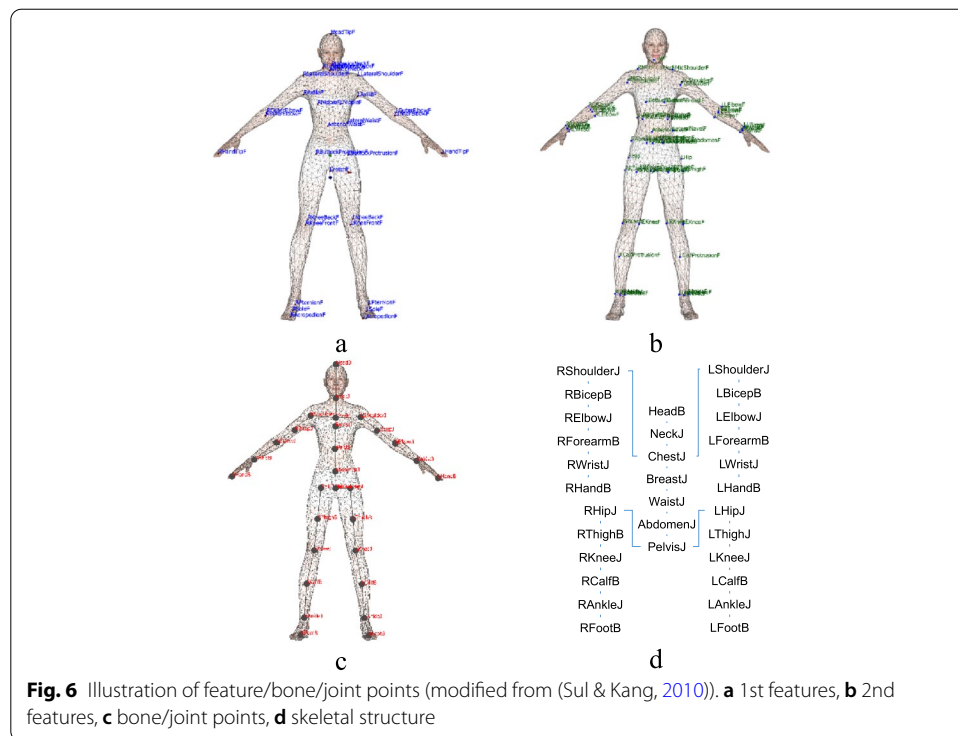
$$N_s = 1 + \text{floor}\left(\frac{\|\text{OBB}_s\|_z}{L_b}\right), \quad (6)$$

where function $\text{floor}(x)$ returns the maximum integer smaller than xx and $\|\text{OBB}_s\|_z$ represents the OBB size of mesh S with respect to the global z-axis.

Consider that a printer (for example, the Sindoh DP-202 printer used in this study) has an L_b of 18.0 cm while the vertical size of the object is 65.0 cm; in this case, if the V-CUT needs 4 OBBs, the vertical size of each OBB would be $65.0/4 = 16.25$ cm, as illustrated in Fig. 5a and b.

PIE-CUT

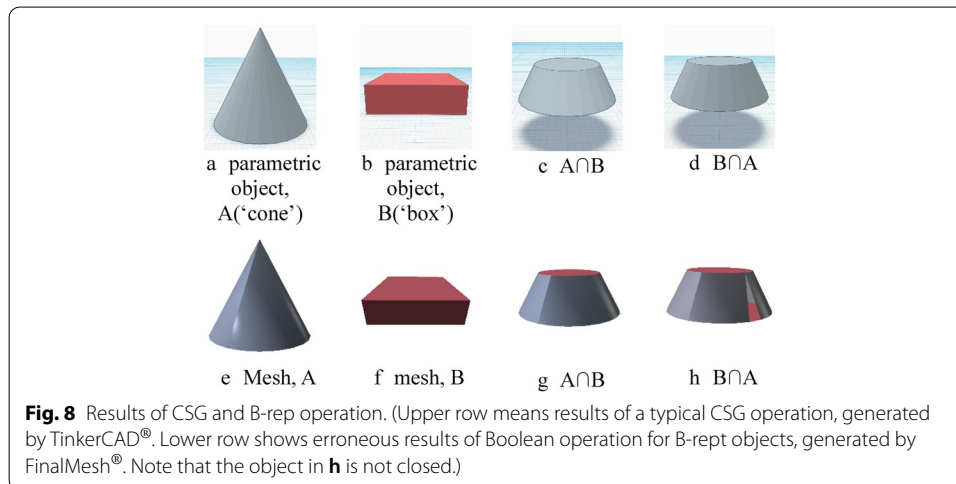
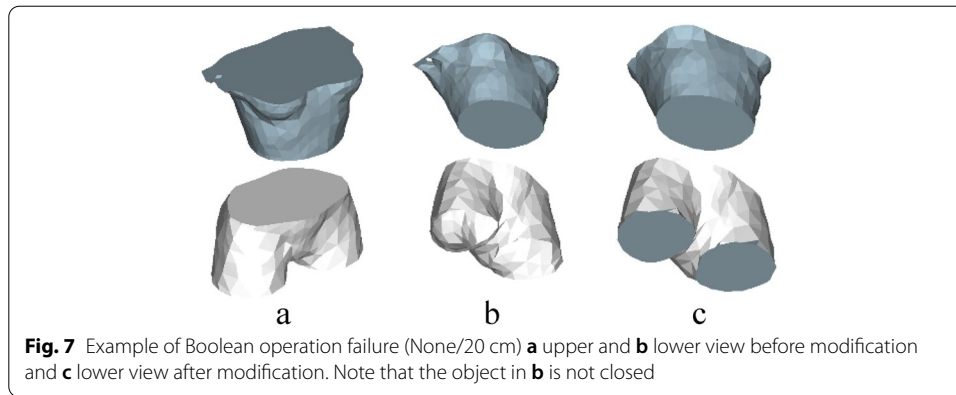
If the object is relatively larger than the printing volume, additional slicing is required in the horizontal direction. The OBBs of the V-CUT mesh (shown as wireframes in Fig. 6b) were dissected into several pie pieces in the circular direction to yield PIE-CUT OBBs (Fig. 5c). The number of pies was controlled so that the largest dimension of these PIE-CUT OBBs is smaller than L_b . After the Boolean operation, V-CUT meshes were converted to PIE-CUT shapes (Fig. 5d).

**Table 1** Nomenclature of feature, bone, and joint points (modified from Sul & Kang, 2010)

Base name	1st Feature points	HeadTip, Neck, Nipple, Waist, Crotch, Elbow, HandTip, Buttock, Knee, Sole
	2nd Feature points	BottomBreast, Shoulder, Abdomen, Hip, Thigh, Ankle, Jaw, AdamsApple
	Bone/Joint points	Head, Neck, Chest, Breast, Waist, Abdomen, Pelvis, Shoulder, Bicep, Elbow, Forearm, Wrist, Hand, Hip, Thigh, Knee, Calf, Ankle, Foot
prefix	L-, R-	Left/right arm or leg
	Lateral-	Nearest point on skin to side/ forward/backward direction
	Anterior- Posterior-	
suffix	-F	1 st or 2 nd feature points
	-B	Bone points (zero degree of freedom)
	-J	Joint points (non-zero degree of freedom)

Body-feature-point extraction

Based on our previous study, feature points and bone/joint points were extracted to specify the bounding boxes' size and position (Sul & Kang, 2010). Table 1 shows the nomenclature of the extracted points. Figure 6 show the 1st and 2nd feature points and bone/joint points for our manikin mesh data. The 1st feature points (Fig. 6a) can be directly found from the mesh surface, such as hand tip, crotch, head tip, and so on. Meanwhile, the 2nd feature points (Fig. 6b) are indirectly found from the 1st feature points and the bone/joint points. Bone/joint points (Fig. 6c) compose the skeleton structure (Fig. 6d) inside the manikin body, while two kinds of feature points lie on the manikin mesh surface.



Results and discussion

Modified Boolean operator

Theoretically, a Boolean operation should always generate a closed-volume mesh. However, we observed that this was not the case, as illustrated in Fig. 7a and b. Note that the bottom faces of both thighs are missing. Such non-closed volumes, also known as non-manifold models in CAD, are detrimental to successful 3D printing. This is due to the incompatibility between the two mesh display methods, boundary representation (B-rep) and CSG. Most graphics or CAD engines display data using the B-rep method, which expresses shapes by vertex, edge, and face information. Meanwhile, the Boolean operation is essential for objects with CSG, in which objects are defined parametrically. Autodesk TinkerCAD[®] is one particular CSG-type modeling software. In Fig. 8, a Boolean operation between the parametric shapes, “cone” (Fig. 8a) and “box” (Fig. 8b), generated closed volume in any operand order (Fig. 8c and d). These objects were again exported to B-rep data in Wavefront[®].OBJ format, as shown in Fig. 8e and f. Boolean operator was applied using commercial software (FinalMesh[®]) afterward; however, this operation failed sometimes (Fig. 8h). We call this phenomenon the “CSG/B-rep

```

Input data: mesh  $A(<\mathbf{f}_m, \mathbf{n}_m>)$ ,  $B(<\mathbf{f}_o, \mathbf{n}_o>)$ 

Begin Function

  Merge vectors of same direction in
   $<\mathbf{n}_m>$  and call it  $<\mathbf{n}_n>$ 
  Collect faces whose normal vector
  belongs to  $<\mathbf{n}_n>$  and call it  $<\mathbf{f}_n>$ 
  Create a new mesh  $A_{new}(<\mathbf{f}_n', \mathbf{n}_n'>)$ 

  Merge vectors of same direction in
   $<\mathbf{n}_o>$  and call it  $<\mathbf{n}_p>$ 
  Collect faces whose normal vector
  belongs to  $<\mathbf{n}_p>$  and call it  $<\mathbf{f}_p>$ 
  Create a new mesh  $B_{new}(<\mathbf{f}_p', \mathbf{n}_p'>)$ 

  Calculate plane-to-plane distance
   $<d_{np}>$  between
  faces of  $A_{new}(<\mathbf{f}_n', \mathbf{n}_n'>)$  and  $B_{new}(<\mathbf{f}_p', \mathbf{n}_p'>)$ 

  For each  $d_{ik}$  in  $<d_{np}>$ 
    If  $d_{ik} = 0$ ,
      Translate vertices of face  $\mathbf{f}_k$  of mesh
       $B$  by  $\theta \cdot \mathbf{n}_k$ 
      (where  $\theta$  = predefined value)
    End of If
  End of For

  Execute Boolean operation between  $A$ 
  and  $B$ 

End Function

```

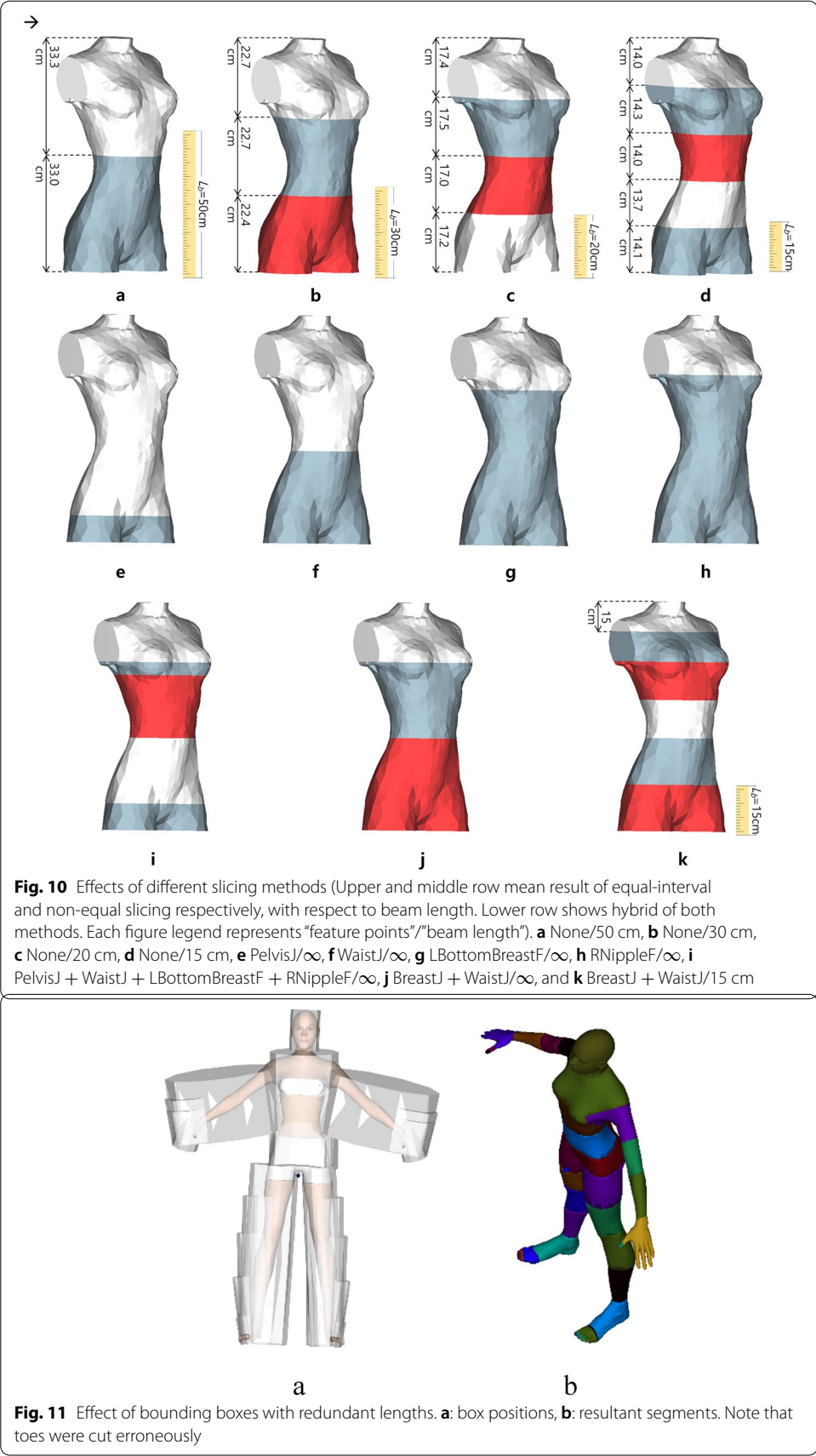
Fig. 9 Pseudocode of the modified Boolean operation

incompatibility problem.” To resolve this problem, we propose a modified Boolean operator, as shown in Fig. 9. The basic idea is that when any face of mesh B meets any face of mesh A, it is lifted by a certain distance θ . As a high value of θ can affect the operation result, it was fixed at 10^{-5} cm in this study.

Verification of the proposed method using bodice slicing

The manikin data (Fig. 4a) was segmented using the 3 steps described in “Modeling” section and the modified Boolean operator. Figure 10a~d show equal-interval slicing results when the manikin bodice’s vertical size was 67.8 cm, and L_b was 50, 30, 20, and 15 cm. It shows that our bounding-box-based approach can successfully partition large objects. Meanwhile, erroneous slicing can occur if the bounding box’s size is not accurate (Fig. 11a). In Fig. 11b, the toes were unintentionally segmented due to the large length of the OBB. The same result can occur if a cutting plane of infinite size is used.

Our method can also generate non-equal-interval slicing. The user can easily change the size of the existing OBB simply by changing its min/max span values. This can be done more easily using feature/bone/joint positions, as shown in Fig. 12. The same method was used on the manikin bodice data, using the feature-point information in Fig. 6. Figure 10e~h show the results of feature-point-based non-equal-interval slicing at $L_b = \infty$ (In the actual implementation, a large number such as 100 cm was used instead of infinity). In Fig. 10e~h, only a single feature point was used to



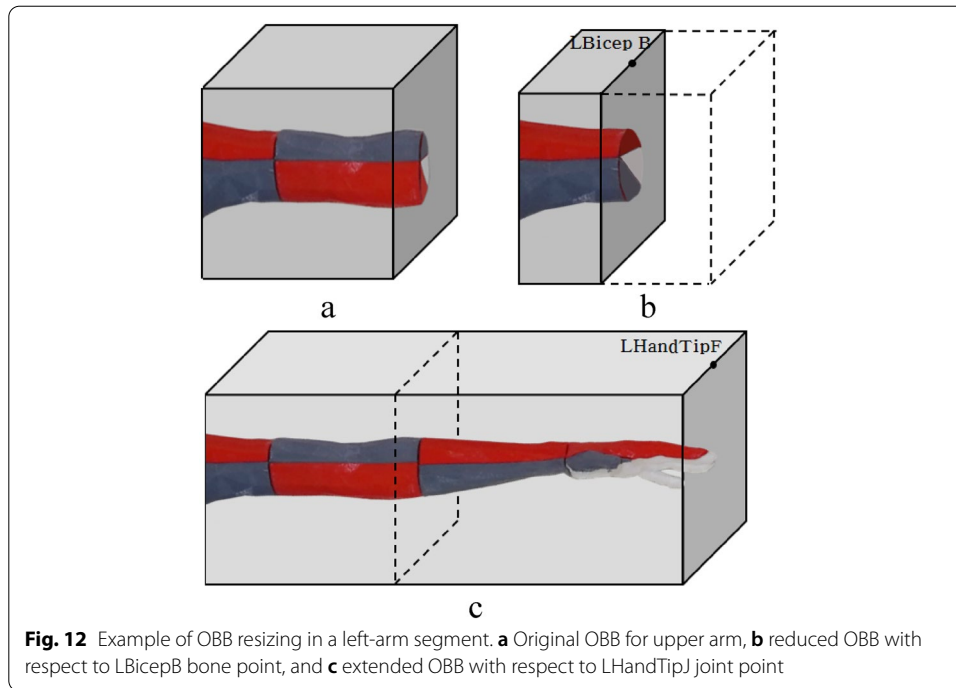


Fig. 12 Example of OBB resizing in a left-arm segment. **a** Original OBB for upper arm, **b** reduced OBB with respect to LBicepB bone point, and **c** extended OBB with respect to LHandTipJ joint point

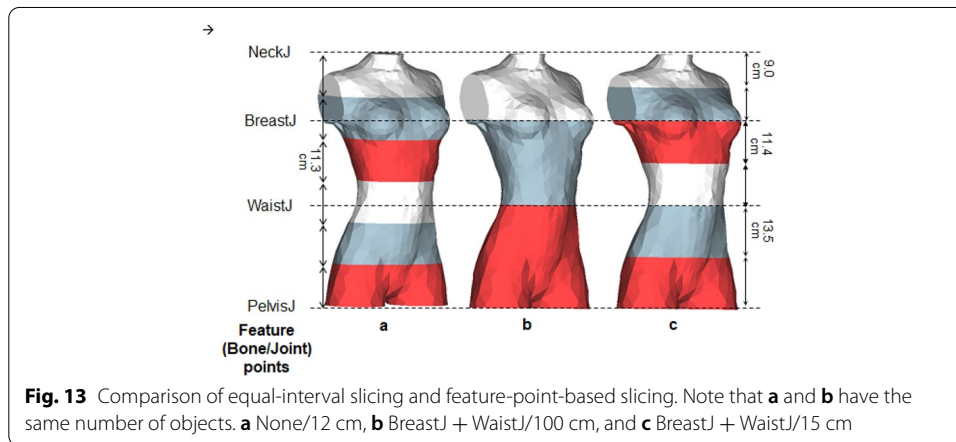


Fig. 13 Comparison of equal-interval slicing and feature-point-based slicing. Note that **a** and **b** have the same number of objects. **a** None/12 cm, **b** BreastJ + WaistJ/100 cm, and **c** BreastJ + WaistJ/15 cm

change the size of the OBB; multiple feature points can also be used on a single OBB as shown in Fig. 10i and j, in which 4 feature points (PelvisJ, WaistJ, LBottomBreastF, and RNippleF) and 2 points (BreastJ and WaistJ) were used at $L_b = \infty$. Furthermore, the two kinds of slicing can be conducted simultaneously. Figure 10k used the same feature-point constraint as Fig. 10j, while $L_b = 15$ cm. It can be clearly seen that the three parts of Fig. 10j were segmented again with equal spacing while maintaining the original dashed slicing line around BreastJ and WasitJ. Figure 13 quantitatively compares equal-interval slicing and feature-point-based slicing. Figure 13b and c are similar to Fig. 10j and k, respectively, while Fig. 13a depicts equal-interval slicing at $L_b = 12$ cm. It is to be noted that Fig. 13a and c contain the same number of parts (8), but with different sizes. Figure 13 implies that the user can have many degrees of freedom in slicing manikin data using the proposed method.

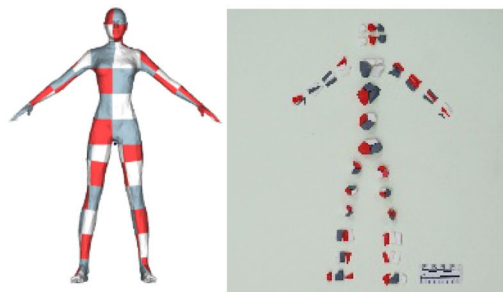
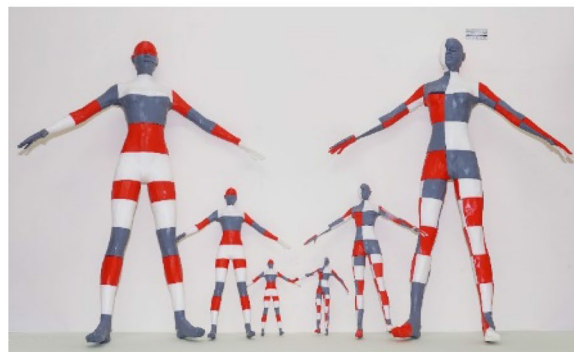
**a****b****c**

Fig. 14 Screenshot of the final 3D-printed manikins. **a** V-CUT simulation (left) and 1/8 scale printed result (right) without any post-processing, **b** PIE-CUT simulation (left) and 1/8 scale printed result (right) without any post-processing, **c** from left to right, 1/2, 1/4, and 1/8 scale glued manikins from V-CUT and 1/8, 1/4, and 1/2 scale ones from PIE-CUT, respectively

Slicing of A-posed whole-body manikin

In “[Verification of the proposed method using bodice slicing](#)” section, our method was verified by slicing the bodice part. In this section, we shall describe the segmentation and 3D printing of whole-body manikin data. Figure 14a shows the results of equal-interval slicing at $L_b = 18$ cm. In Fig. 14b, the resultant parts have three PIE-CUTs for each layer, but the number of pies can vary depending on the L_b value. Although the manikin’s original size was 173.1 cm, only 1/8, 1/4, and 1/2 scales were actually printed



Fig. 15. 3D printing of 1/8 scale manikin without proper slicing (Kim & Sul, 2018)

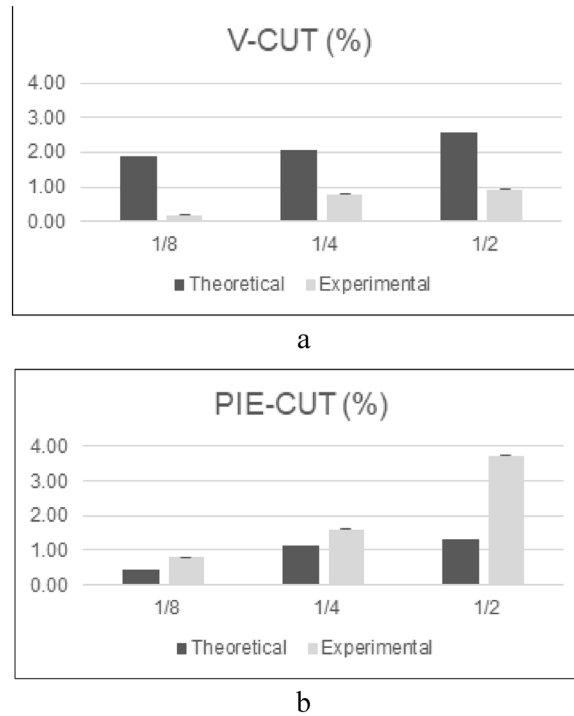
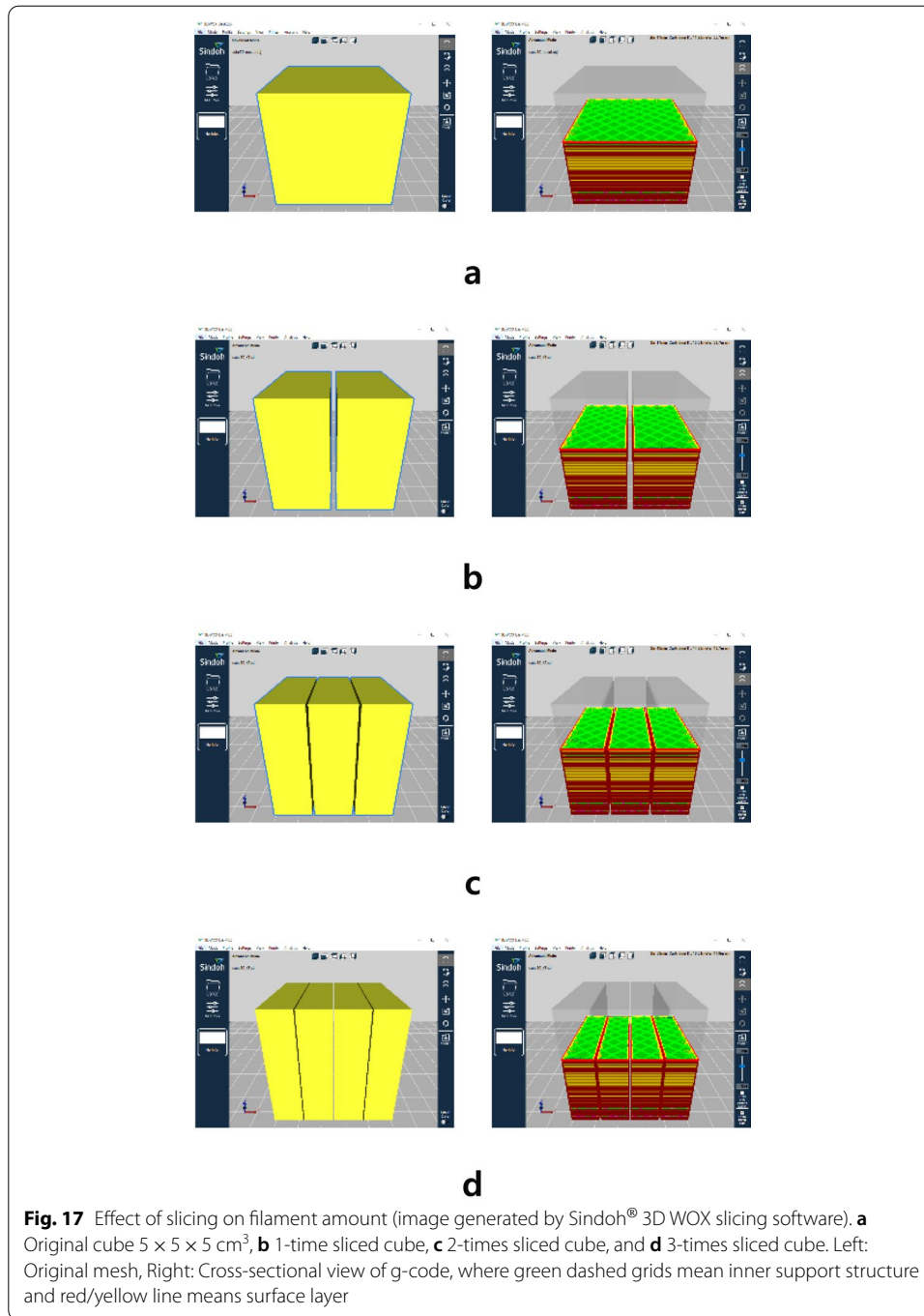


Fig. 16 Quantitative measurement of R_{ss} for two kinds of slicing. **a** V-CUT and **b** PIE-CUT

to conserve printing time. And then, each segment was glued to make a final manikin. The left-hand side and right-hand side figures of Fig. 14c show the results of V-CUT and PIE-CUT operations, respectively. A scale bar with dimensions of $5.0 \times 2.5 \text{ cm}^2$ is inserted in Fig. 14a ~ c to show clearly the printed parts' actual size.

Figure 14c–h also show that there were few support structures in the final product. On the contrary, in our previous study, in which slicing was done manually using commercial 3D mesh operation software, 1/8 scale manikin printing resulted in 63.6% (Fig. 15a) of waste (Kim & Sul, 2018). We defined “filament waste ratio (R_{ss})” due to the support structure in 3D printing using Eq. (7) for quantitative comparison.



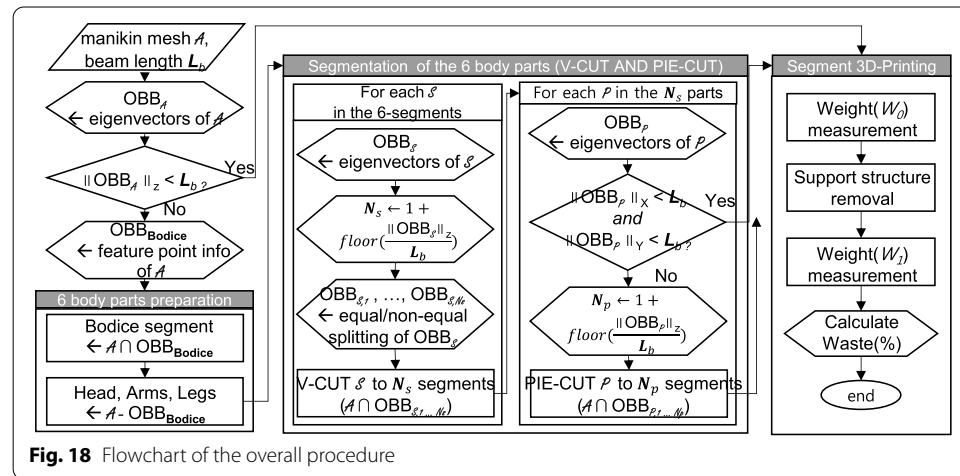
$$R_{ss}(\%) = \frac{W_0 - W_1(g)}{W_1(g)} \times 100(\%) = \frac{\text{Weight of support structure}(g)}{\text{Weight of manikin}(g)} \times 100(\%) \quad (7)$$

Here, W_0 and W_1 are the total weights of 3D-printed parts before/after support-structure removal, respectively. Figure 16 shows the R_{ss} values of V-CUT and PIE-CUT results; filament wastage was less than 1.0% in V-CUT and 4.0% in PIE-CUT. This means that our method is more environmentally friendly as it reduces the

Table 2 Measurement of calculation speed

Type	Operation [†]	# of segments generated	Avg. Calc. time (seconds)
6 body part preparation	OBB	6	0.144
	Bool		0.633
V-CUT	OBB	26	0.386
	Bool		1.685
PIE-CUT	OBB	78	0.609
	Bool		2.818
sum			6.275

[†] : "OBB" and "Bool" means OBB preparation and Modified Boolean operation respectively

**Fig. 18** Flowchart of the overall procedure

amount of plastic wastage. It should be noted that the experimental PIE-CUT values are greater than the theoretical values, while V-CUT results showed the opposite tendency. As more slicing was done, there were no significant changes in the inner and outer support structures' total volume. However, two new boundary faces are created beside the slicing line, which can be thought to be a filament wastage source. In most 3D-printer-slicing software, unlike the outer surface, the support structure's inner surface can be controlled or reduced to preserve the input object's shape. This fact can be easily verified using commercial slicing software. Figure 17 shows a cube of $5 \times 5 \times 5 \text{ cm}^3$, and its 1, 2, or 3 times sliced objects in the x-axis direction were converted to g-code. Although the four objects have the same volume, the amount of filament needed for 3D printing varied at 11.96, 13.31, 14.66, and 16.01 m. Several studies attempted to predict the optimal printing direction or support-structure-fill pattern to reduce the amount of total filament required (Ezair et al. 2015; Jiang et al. 2018; Wang et al. 2013, 2017), but few studies focused on the effect of slicing in 3D printing. We shall focus on this aspect in the near future.

The overall calculation speed is also important to users. Our method processes input mesh data in only 2 passes (V-CUT and PIE-CUT), unlike the 300,000 passes described in previous studies (Nicholson, 2016). Using the proposed approach, it took

Table 3 Amount of filament length and time required for 3D printing (unit means millimeter/hour:minute)

	1/8 scale	1/4 scale	1/2 scale
V-CUT	8.98/3:30	49.71/15:33	298.99/90:57
PIE-CUT	11.38/5:26	62.2/35:19	349.73/109: 1

only ~6 s to generate the parts shown in Fig. 14a and b on an MS Windows 10[®] based PC (AMD Ryzen[®] 7 2700 × 3.70 GHz 8-core processor), as shown in Table 2. Figure 18 shows a flowchart of the proposed method.

The proposed method is versatile for most 3D-scanned human body data or graphically generated manikin mesh data. However, it is limited by its condition that the pose should be a standard A-pose. The reason for imposing this limitation on the manikin pose was that the human-feature point-detection algorithm could work only with A-pose data. If the feature/bone/joint-point information can be input manually or using other methods, our algorithm can be used on manikins of arbitrary poses.

Furthermore, our method successfully reduced filament waste by slicing raw data. As shown in Table 3, it took several days to print a half-size manikin. Therefore, converting raw mesh data to a shell structure is expected to reduce the inner support structure.

Conclusions

In summary, an optimized slicing algorithm was proposed for manikins with Boolean operations between the input manikin mesh data and bounding boxes. The bounding box can easily specify the region to print. The bounding boxes' size and directions were automatically calculated using body feature/bone/joint information by PCA. Using this simple box-based geometry, the proposed method can be easily adopted in user-friendly open-source tools like Python, MATLAB, etc. Nevertheless, the B-rep based mesh data and CSG-based Boolean operation belong to a different category, and their combination resulted in erroneous cutting. The numerical round-off error frequently occurred when the two operand meshes have the same plane. We designated this phenomenon the “CSG/B-rep incompatibility problem” and it was resolved by modifying the Boolean operator. Once the bounding boxes and modified Boolean operators were ready, a manikin's actual slicing operation was carried out in 3 steps. The initial A-pose manikin was decomposed into 6 segments, viz. the head, bodice, arms, and legs. Each segment was then processed by V-CUT and PIE-CUT operations if the decomposed segment was bigger than the predefined printer size (beam length). The size of the OBBs could be controlled by equal-interval or non-equal-interval slicing, depending on the beam length. Besides, the user can specify the slicing position freely with feature-point information using the proposed approach. The proposed method was successfully applied on a bodice segment and whole-body mesh data, and it was found that filament waste could be reduced to less than 4.0%. These operations could be carried out in only a few seconds on a personal computer as the proposed method processes the whole manikin data only twice, unlike previously described methods. However, the total filament amount and printing time were too large, even for a half-size manikin. Making the manikin volume into a shell-like structure could

decrease the amount of internal structure. Also, the input manikin could have only an A-shape due to automatic feature-point detection. In our future studies, we shall work toward making this approach more generalized.

Acknowledgements

The authors are grateful to former graduate student Ms. Ji Young Jeong for her dedicated participation in project management. Also, this work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIT) (No. 2019R1F1A1040883).

Authors' contributions

JYJ conducted experiments and analyzed the results with SC and IHS. SC collected literature and developed theoretical models. IHS developed programming source codes. All authors read and approved the final manuscript.

Funding

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2019R1F1A1040883).

Availability of data and materials

The datasets used and analyzed during the current study are available from the corresponding author on reasonable request.

Declarations

Competing interests

The authors declare that they have no competing interests.

Author details

¹ Graduate Student, Dept. of Materials Design Engineering, Kumoh National Institute of Technology, Gumi 39177, Republic of Korea. ² Associate Professor, Dept. of IT Convergence, Kumoh National Institute of Technology, Gumi 39177, Republic of Korea. ³ Associate Professor, Dept. of Materials Design Engineering, Kumoh National Institute of Technology, Gumi 39177, Republic of Korea.

Received: 29 October 2020 Accepted: 3 March 2021

Published online: 25 March 2021

References

- Bos, F., Wolfs, R., Ahmed, Z., & Salet, T. (2016). Additive manufacturing of concrete in construction: potentials and challenges of 3D concrete printing. *Virtual and Physical Prototyping*, 11(3), 209–225. <https://doi.org/10.1080/17452759.2016.1209867>
- Boström, O., Fredriksson, R., Håland, Y., Jakobsson, L., Krafft, M., Lövsund, P., Muser, M. H., & Svensson, M. Y. (2000). Comparison of car seats in low speed rear-end impacts using the BioRID dummy and the new neck injury criterion (NIC). *Accident Analysis & Prevention*, 32(2), 321–328. [https://doi.org/10.1016/S0001-4575\(99\)00105-0](https://doi.org/10.1016/S0001-4575(99)00105-0)
- Chen, X., Liu, X., Quyang, M., Chen, J., Taiwo, O., Xia, Y., Childs, P. R., Brandon, N. P., & Wu, B. (2019). Multi-metal 4D printing with a desktop electrochemical 3D printer. *Scientific reports*, 9(1), 1–9. <https://doi.org/10.1038/s41598-019-40774-5>
- Chen, X., Zhang, H., Lin, J., Hu, R., Lu, L., Huang, Q.-X., Benes, B., Cohen-Or, D., & Chen, B. (2015). Dapper: decompose-and-pack for 3D printing. *ACM Transactions on Graphics*, 34(6), 213:211–213:212. <https://doi.org/10.1145/2816795.2818087>
- Crump, S. S. (1992). *Apparatus and method for creating three-dimensional objects* (US Patent No. 5,121,329). U. S. Patent and Trademark Office.
- Deckard, C. R. (1989). *Method and apparatus for producing parts by selective sintering* (US Patent No. 4,863,538). U. S. Patent and Trademark Office.
- Demir, İ., Aliaga, D. G., & Benes, B. (2018). Near-convex decomposition and layering for efficient 3D printing. *Additive Manufacturing*, 21, 383–394. <https://doi.org/10.1016/j.addma.2018.03.008>
- Duran, C., Subbian, V., Giovanetti, M. T., Simkins, J. R., & Beyette, F. R., Jr. (2015). Experimental desktop 3D printing using dual extrusion and water-soluble polyvinyl alcohol. *Rapid Prototyping Journal*. <https://doi.org/10.1108/rpj-09-2014-0117>
- Ezair, B., Massarwi, F., & Elber, G. (2015). Orientation analysis of 3D objects toward minimal support volume in 3D-printing. *Computers & Graphics*, 51, 117–124. <https://doi.org/10.1016/j.cag.2015.05.009>
- International Organization for Standardization. (2017). *Size designation of clothes. Part 1: Anthropometric definitions for body measurement* (ISO Standard No. 8559–1: 2017). <https://www.iso.org/standard/61686.html>
- International Organization for Standardization. (2015). *Additive Manufacturing—General Principles—Terminology* (ISO/ASTM Standard No. 52900: 2015). <https://www.iso.org/standard/69669.html>
- Jadoon, A. K., Wu, C., Liu, Y.-J., He, Y., & Wang, C. C. (2018). Interactive partitioning of 3D models into printable parts. *IEEE Computer Graphics and Applications*, 38(4), 38–53. <https://doi.org/10.1109/mcg.2018.182130719>
- Jiang, J., Xu, X., & Stringer, J. (2018). Support structures for additive manufacturing: a review. *Journal of Manufacturing and Materials Processing*, 2(4), 64. <https://doi.org/10.3390/jmmp2040064>

- Kim, S., & Sul, I. H. (2018). Minimization of support structure using cone-shaped body segmentation in fused deposition modeling type small size human manikin 3D printing. *Textile Science and Engineering*, 55(3), 194–201. <https://doi.org/10.12772/TSE.2018.55.194>.
- Lu, L., Sharf, A., Zhao, H., Wei, Y., Fan, Q., Chen, X., Savoye, Y., Tu, C., Cohen-Or, D., & Chen, B. (2014). Build-to-last: strength to weight 3D printed objects. *ACM Transactions on Graphics (ToG)*, 33(4), 1–10. <https://doi.org/10.1145/2601097.2601168>
- Luo, L., Baran, I., Rusinkiewicz, S., & Matusik, W. (2012). Chopper: partitioning models into 3D-printable parts. *ACM Transactions on Graphics*, 31(6), 129. <https://doi.org/10.1145/2366145.2366148>
- Murphy, S. V., & Atala, A. (2014). 3D bioprinting of tissues and organs. *Nature Biotechnology*, 32(8), 773. <https://doi.org/10.1038/nbt.2958>
- Nayak, R., & Padhye, R. (2017). *Manikins for textile evaluation*. Woodhead Publishing.
- Ngo, T. D., Kashani, A., Imbalzano, G., Nguyen, K. T., & Hui, D. (2018). Additive manufacturing (3D printing): A review of materials, methods, applications and challenges. *Composites Part B: Engineering*, 143, 172–196. <https://doi.org/10.1016/j.compositesb.2018.02.012>
- Nicholson, M. (2016). *QChopper-segmentation of large surface meshes for 3D printing* [Unpublished doctoral dissertation]. Canada: Queen's University.
- PelikanSoftwareKft. (2020). *FinalMesh Boolean API*. http://www.finalmesh.com/boolean3d_api.htm. Retrived from, October 26, 2020.
- Pentland, A., & Williams, J. (1989). Good vibrations: Modal dynamics for graphics and animation. In *Proceedings of the 16th Annual Conference on Computer Graphics and Interactive*, vol 23 (pp. 215–222). <https://doi.org/10.1145/74333.74355>.
- Perrot, A., Rangeard, D., & Pierre, A. (2016). Structural built-up of cement-based materials used for 3D-printing extrusion techniques. *Materials and Structures*, 49(4), 1213–1220. <https://doi.org/10.1617/s11527-015-0571-0>.
- Shen, Z.-H., Dai, N., Li, D.-W., & Wu, C.-Y. (2016). Bridge support structure generation for 3D printing. *2015 international workshop on materials, manufacturing technology, electronics and information science* (pp. 141–149). https://doi.org/10.1142/9789813109384_0016.
- Stromquist, W. (1980). How to cut a cake fairly. *The American Mathematical Monthly*, 87(8), 640–644. <https://doi.org/10.2307/2320951>
- Sul, I. H., & Kang, T. J. (2010). Regeneration of 3D body scan data using semi-implicit particle-based method. *International Journal of Clothing Science and Technology*, 22(4), 248–271. <https://doi.org/10.1108/09556221011048286>
- Talataisong, W., Ismaeel, R., Marques, T. H., Mousavi, S. A., Beresna, M., Gouveia, M., Sandoghchi, S. R., Lee, T., Cordeiro, C. M., & Brambilla, G. (2018). Mid-IR Hollow-core microstructured fiber drawn from a 3D printed PETG preform. *Scientific reports*, 8(1), 1–8. <https://doi.org/10.1038/s41598-018-26561-8>
- Teschner, M., Kimmmerle, S., Heidelberger, B., Zachmann, G., Raghupathi, L., Fuhrmann, A., Cani, M. P., Faure, F., Magnenat-Thalmann, N., & Strasser, W. (2005). Collision detection for deformable objects. *Computer Graphics Forum*. <https://doi.org/10.1111/j.1467-8659.2005.00829.x>
- Thielen, M., & Delbressine, F. (2016). Rib cage recreation: Towards realistic neonatal manikin construction using MRI scanning and 3D printing. In *Proceedings of SMI'2016 fabrication and sculpting event (FASE)*, Germany (pp. 20–23).
- Vaidya, R., & Anand, S. (2016). Optimum support structure generation for additive manufacturing using unit cell structures and support removal constraint. *Procedia Manufacturing*, 5, 1043–1059. <https://doi.org/10.1016/j.promfg.2016.08.072>
- Vanek, J., Galicia, J., Benes, B., Měch, R., Carr, N., Stava, O., & Miller, G. (2014). Packmerger: A 3d print volume optimizer. *Computer Graphics Forum*. <https://doi.org/10.1111/cgf.12353>
- Wang, W., Chao, H., Tong, J., Yang, Z., Tong, X., Li, H., Liu, X., & Liu, L. (2015). Saliency-preserving slicing optimization for effective 3D printing. *Computer Graphics Forum*. <https://doi.org/10.1111/cgf.12527>
- Wang, W., Liu, Y.-J., Wu, J., Tian, S., Wang, C. C., Liu, L., & Liu, X. (2017). Support-free hollowing. *IEEE Transactions on Visualization and Computer Graphics*, 24(10), 2787–2798. <https://doi.org/10.1109/tvcg.2017.2764462>
- Wang, W., Wang, T. Y., Yang, Z., Liu, L., Tong, X., Tong, W., Deng, J., Chen, F., & Liu, X. (2013). Cost-effective printing of 3D objects with skin-frame structures. *ACM Transactions on Graphics (ToG)*, 32(6), 177. <https://doi.org/10.1145/2508363.2508382>
- Wei, X., Qiu, S., Zhu, L., Feng, R., Tian, Y., Xi, J., & Zheng, Y. (2017). Toward support-free 3d printing: A skeletal approach for partitioning models. *IEEE Transactions on Visualization and Computer Graphics*, 24(10), 2799–2812. <https://doi.org/10.1109/TVCG.2017.2767047>
- Zhang, X., Xia, Y., Wang, J., Yang, Z., Tu, C., & Wang, W. (2015). Medial axis tree—an internal supporting structure for 3D printing. *Computer Aided Geometric Design*, 35, 149–162. <https://doi.org/10.1016/j.cagd.2015.03.012>

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.