Doctoral Thesis

Developing a Novel Reverse Engineering Method Suitable for 3D Printing and its Application to Preserving Cultural Heritage

（三次元プリンティングに適した新リバースエンジニアリング手法の開発とその文化遺産保存への応用）

March 2021

Tashi

Doctoral Thesis

Developing a Novel Reverse Engineering Method Suitable for 3D Printing and its Application to Preserving Cultural Heritage

（三次元プリンティングに適した新リバースエンジニアリング手法の開発とその文化遺産保存への応用）

March 2021

Tashi

Abstract

Reverse engineering mimics critical features of a given object and creates its accurate (or enhanced) virtual model (digital model) and replica (physical model). It is often used to digitize existing objects in different engineering fields and preserve significant cultural artifacts.

In conventional reverse engineering, a 3D scanner scans the exposed surfaces of a given object, which results in a relatively large point cloud. The noise and outlier points are removed from the point cloud before the surfaces are reconstructed. The reconstructed surfaces undergo geometric modeling, which results in a digital model of the object. This kind of reverse engineering requires sophisticated devices (a 3D scanner) and complex computations. It is challenging to make a reverse engineering process less dependent on sophisticated devices and complex computations.

This study proposed a novel reverse engineering method to overcome the challenges. The proposed novel reverse engineering method uses an analytical approach to create a noise- and outlier-free point cloud of a given object. The analytical approach is based on a recursive process that requires two types of input. The first type of input consists of two parameters denoted as instantaneous distance and instantaneous rotational angle. These parameters vary in each iteration of the recursive process. The other input of the recursive process consists of three parameters: center point, initial distance, and initial angle. They remain constant for all iterations. The recursive process produces a small point cloud to model some features of a given object, even when the mathematical representations are unknown. In this case, the instantaneous rotational angle can be increased linearly after each iteration, and the instantaneous distance can be varied using simple and well-known mathematical functions.

The modeling ability of the proposed reverse engineering method is validated by comparing two point clouds of the same shape. The first point cloud is created using the parametric equation for a given shape (e.g., circle, ellipse, spiral, astroid, and straight lines). The second point cloud of the same shape is created using the recursive process.

Significant cultural artifacts can be digitized using the proposed reverse engineering method. This study considers reverse engineering some patterns belonging to the Ainu, the indigenous people of Hokkaido. They use fourteen elementary motifs (e.g., "Ayus, "

"Morew, " "Sik," and "Utasa") to create unique patterns with which they decorate their houses, clothing, ornaments, utensils, and spiritual goods. The mathematical representations of these motifs and the underlying patterns are unavailable. The proposed reverse engineering method is applied to create point clouds representing some selected Ainu motifs. For each point cloud, the instantaneous rotational angle is increased linearly and the instantaneous distance is adjusted as needed using some simple mathematical functions. This way, a database of Ainu motifs is prepared. Some point clouds are exported to a commercially available 3D CAD system. After simple geometric modeling (rotation, translation, extrusion, and copying), the point clouds are transformed into digital models (virtual models) of the respective motifs and patterns. The digital models are used to manufacture replicas of the respective motifs/patterns with an ordinary 3D printer. This way, this study digitally preserves the craftsmanship of Ainu motifs and patterns.

The proposed reverse engineering method equally applies to artifacts other than Ainu motifs. It is demonstrated by producing a virtual model and replica of an ancient ewer (i.e., a 3D shape). The processes involved in the proposed reverse engineering method (recursive point cloud creation process, point cloud coordination, solid CAD modeling, and replica manufacturing using a 3D printer) do not require sophisticated devices. They are free from heavy computations, unlike conventional reverse engineering processes.

The thesis is organized into six chapters. The first chapter presents the background, scope, objectives, and contribution of this study. The second chapter provides a literature review on reverse engineering, its application to 3D printing, and Ainu motifs/patterns. The third chapter describes the methodology of this study. The fourth chapter details applications of the proposed reverse engineering method, particularly for creating a virtual/physical prototype of Ainu motifs/patterns and other significant cultural artifacts. The fifth chapter discusses the implications of this study and highlights future work. The final chapter concludes this thesis.

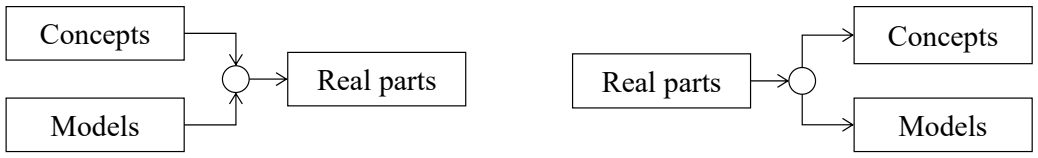
Table of Contents

Chapter 1: Introduction

This chapter addresses the introduction of this study. The introduction outlines the background (Section 1.1), the context (Section 1.2), the scope (Section 1.3), the aim and objectives (Section 1.4), and the contribution and significance (Section 1.5) of this study. Finally, Section 1.6 presents the structure of this thesis.

1.1. Background

Nowadays, reverse engineering (RE) [1-3] has earned a great deal of attention due to the advances in 3D scanning [4] and photogrammetry [5] technology. RE is a process to deconstruct an existing object to disclose its designs, architecture, or extract knowledge to reconstruct a digital model [6]. Reverse engineering transforms real parts into engineering concepts and models. In forward engineering, engineering concepts and models are transformed into real parts [7]. A fundamental concept of forward engineering and reverse engineering is illustrated in Figure 1-1.



(*a*) Forward engineering (*b*) Reverse engineering

Figure 1-1. A fundamental concept of forward engineering and reverse engineering.

There are several benefits of reverse engineering [7, 8]. Some of the benefits are listed as follows.

- Create a replica when original products are no longer manufactured or do not have documents or drawings or a real model exists as clay, wood, or form model.
- Modify the previous design, inspection, or analysis to make design improvements.
- Generate custom fits to the human body (e.g., helmet, spacesuit, and prosthesis).

As a result, reverse engineering is deployed in a wide range of engineering fields. For example, biomedical engineering [9], chemical engineering [10], civil engineering

[11], electronics engineering [12], software engineering [6], and mechanical engineering [13]. Reverse engineering is not only in the engineering fields but also in preserving cultural heritage [14, 15].

Reverse engineering can be classified as shape reverse engineering, material reverse engineering, and process reverse engineering. The choice of a reverse engineering approach depends on the area of applications. This study is interested in applying reverse engineering for digitizing significant cultural objects and the physical model realization using the digital manufacturing processes. In this regard, shape reverse engineering is the preferred approach. Shape reverse engineering [7, 13, 16, 17] is often referred to as geometric reverse engineering [7] or computer-aided design (CAD) reverse engineering [18]. The ultimate goal of shape reverse engineering is to reconstruct a virtual model as close as possible to a given model. Digital data of the virtual model is used as a blueprint for producing a physical model (replica). The physical model can be built using digital manufacturing processes, such as additive manufacturing (AM) [19-21] (e.g., 3D printing [22, 23]), subtractive manufacturing (e.g., CNC machining), or formative manufacturing (e.g., injection molding).

Typical reverse engineering in practice is shown in Figure 1-2. As seen in Figure 1-2, reverse engineering can be divided into four steps: data acquisition, data preprocessing, surface reconstruction, and virtual model creation. All the steps are elaborated as follows.

In the data acquisition step [7, 24], a 3D scanner scans the targeted surface of the existing physical object, and it generates a relatively large number of points called point clouds. Sometimes the 3D scanner cannot capture at a single scan and requires multiple scans from different orientations, as shown in Figure 1-2.

In the data preprocessing step [25], the three procedures, namely, registration, noise removal, and data reduction, are necessary. All the fragmented point clouds obtained from multiple scans are combined into a single coordinate system in the registration procedure. In the noise removal procedure, noise and outlier (shadow) points are removed from the point cloud. In the data reduction procedure, the redundant points are removed from the point cloud. These procedures are necessary for accurate modeling.

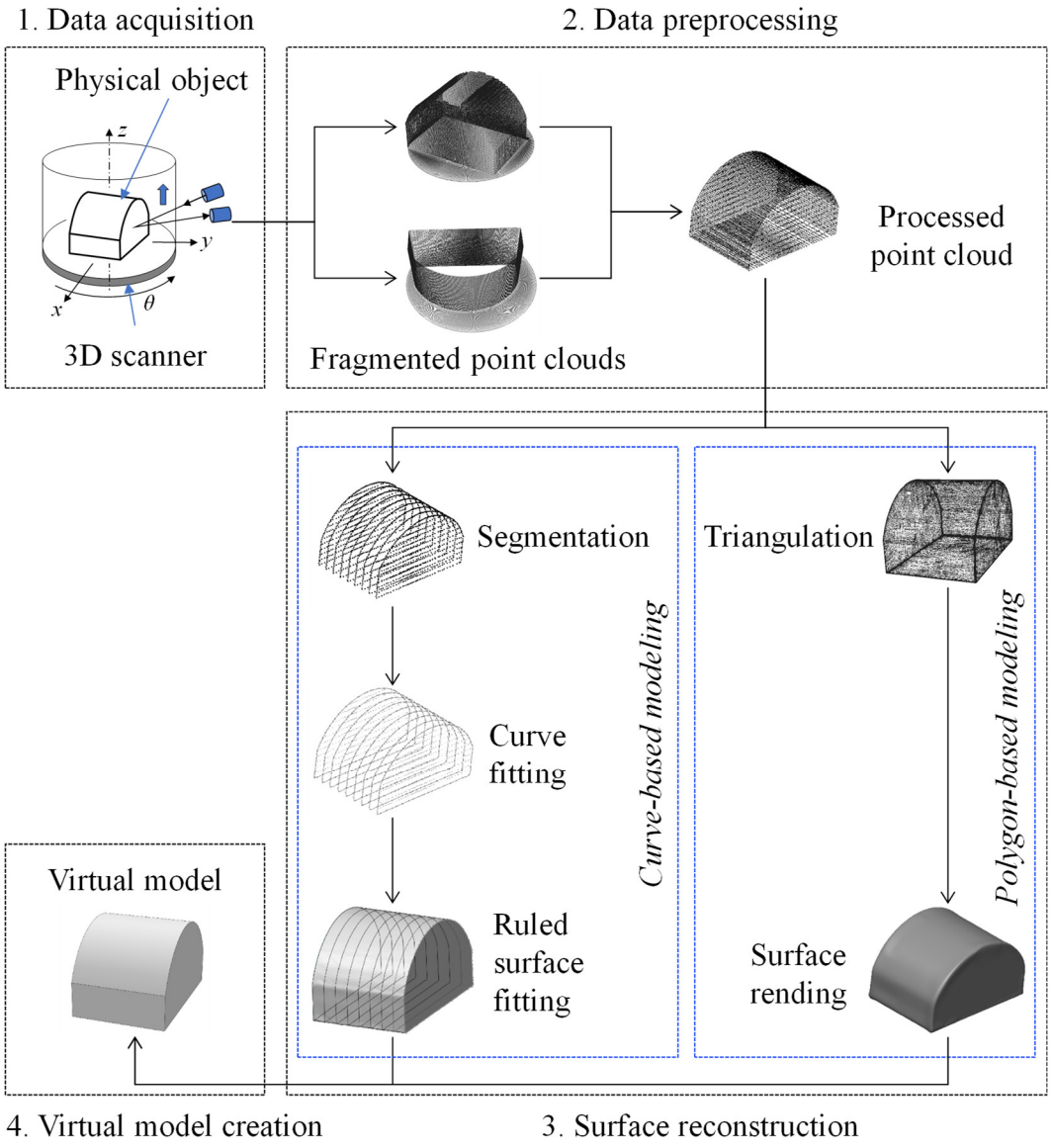


Figure 1-2. Typical reverse engineering in practice.

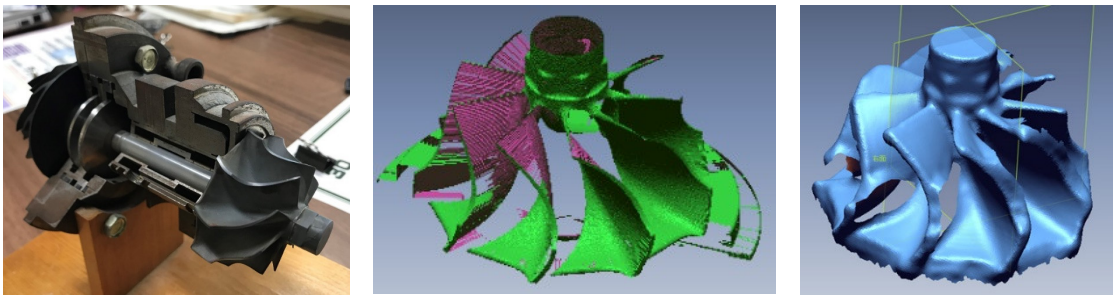
In the surface reconstruction step [26, 27], surfaces are reconstructed from the processed point cloud. Two different geometric modeling approaches, namely, curve-based modeling and polygon-based modeling, are commonly used for surface reconstruction [23]. In the curve-based modeling, the point cloud is divided into subsets into a regular pattern using the segmentation process [28]. The segmented point clouds' cross-sections are fitted using different parametric curves (NURBS, B-splines, and alike) [29]. With the surface fitting, the fitted curves are converted into a ruled or NURBS surface model. In polygon-based modeling, a polyhedral model is directly constructed by connecting neighboring points using one of the following methods: Delaunay triangulation, alpha shaping, crusting, or volumetric. Then it is refined by the decimation

and subdivision algorithms. Next, the surface is rendered on the polyhedral model. The polyhedral model is usually constructed using triangles that connect the points. Thus, polygon-based modeling is also called triangulation modeling.

In the virtual model creation step, the surface model is transferred to a 3D CAD system. Then a virtual model is constructed from the surface model using the functions offered by the 3D CAD system. The digital data of the virtual model is vital for preserving, modification, and manufacturing.

1.2. Context

Conventional reverse engineering requires sophisticated deceives (a 3D scanner) and a 3D CAD system. The process of reverse engineering entirely relies on a 3D scanner and the scanned point cloud. The whole process of reverse engineering can be jeopardized by the accuracy and precision of the 3D scanner itself. Reverse engineering also requires user-intensive geometric modeling and complex computations, such as registration, noise and outlier points removal, redundant points removal, and surface reconstruction. Even though the point clouds are there, it does not guarantee reconstruction of the desired virtual model (CAD model). An example is shown in Figure 1-3. The point cloud is obtained using a stationary Roland 3D scanner, model number PICZA LPX-600 [30], available at the Kitami Institute of Technology. The virtual model reconstructed from the point cloud is not useful for producing a physical model.



(*a*) Turbocharger (*b*) Point cloud obtained by using a 3D scanner (*c*) CAD model after several iterations

Figure 1-3. Example of a 3D scanned point cloud and its CAD model.

Syed et al. [31] reported that surface characteristics affect the point cloud's quality regardless of the scanning accuracy. They reported that the shiny surface and hollow areas are difficult to scan by the 3D scanner. Varady et al. [7] reported the problems associated

with data acquisition, data preprocessing, and surface reconstruction. Geng and Bidhanda [1] and Tashi et al. [32] provided relatively comprehensive issues related to the computational complexity associated with the translation of point clouds into a virtual model. The problems related to reverse engineering is comprehensively reviewed in Chapter 2.

The challenges of the convention reverse engineering arise from the inception of the point cloud itself. If the point cloud can be controlled from the inception, then handling the subsequent process would be eased. One of the pragmatic alternatives is using an analytical point cloud. The advantage of the analytically created point cloud is that the point cloud can be restricted to the small one from inception. The analytical point cloud is free from noise and outlier points. This way, some of the computational complexity above-mentioned can be avoided.

The question is how to create an analytical point cloud useful for modeling a given object? In this regard, there are three main approaches, namely, a deterministic approach [32, 33], a semi-random approach or affine mapping-based approach [34, 35], and a fully random approach [36]. In each case, the information underlined by the analytically created point cloud is used to construct a virtual model systematically. Perhaps this is not the case when the other approaches are used to create the point cloud.

1.3. Scope

The scope of this study is restricted to an analytical point cloud based on a deterministic approach. The deterministic approach uses a recursive process to create an analytical point cloud of the given object. The recursive process requires two types of input. The first type of input consists of two parameters denoted as instantaneous distance and instantaneous rotational angle, which vary in each iteration of the recursive process. The other input of the recursive process consists of three parameters: center point, initial distance, and initial angle, which remain constant for all iterations. The proposed reverse engineering method creates a small point cloud to model the features of the given object, even when the mathematical representations are unknown. In such a case, the instantaneous rotational angle can be increased linearly, and the instantaneous distance can be varied using some well-known mathematical functions.

The modeling ability of the proposed reverse engineering method is validated by comparing two point clouds of the same shape. The first point cloud is created using the parametric equation for a given shape (e.g., circle, ellipse, spiral, astroid, and straight lines). The second point cloud of the same shape is created using the recursive process. When modeling the given object using the proposed reverse engineering method, the following questions are emphasized to ensure modeling consistency and accuracy.

1. What are the role and characteristics of the input parameters (both variable and constants inputs) of the recursive process that creates a point cloud?
2. How to quantify the shape information modeled by a set of a point cloud?
3. Does the analytically created point cloud correctly model a given shape?
4. Does the analytically created point cloud help to create virtual models (solid CAD models) of some given objects?

1.4. Aim and Objectives

This study aims to develop a novel reverse engineering method suitable for 3D printing without using a scanned point cloud. The objectives of this study are as follows:

1. develop a framework of a novel reverse engineering method;
2. develop an analytical point cloud creation process for the proposed reverse engineering method;
3. elucidate the efficacy of the proposed reverse engineering method by digitizing the significant cultural objects; and
4. use 3D printing for prototyping.

1.5. Contribution and Significance

The main contribution of this study is a novel reverse engineering method and its applications for preserving significant cultural artifacts and realization of the physical models using an ordinary 3D printer. The significance of this study is described as follows.

This study considers reverse engineering some patterns belonging to the Ainu, the indigenous people of Hokkaido. They use fourteen elementary motifs (e.g., "Ayus," "Morew," "Sik," and "Utasa") to create unique patterns with which they decorate their

houses, clothing, ornaments, utensils, and spiritual goods. The mathematical representations of these motifs and the underlying patterns are unavailable.

The proposed reverse engineering method is applied to create point clouds representing ten selected Ainu motifs. For each point cloud, the instantaneous rotational angle is increased linearly and the instantaneous distance is adjusted as needed using some simple mathematical functions. This way, a database of Ainu motifs is prepared. Some point clouds are exported to a commercially available 3D CAD system. After simple geometric modeling (rotation, translation, copying, and extrusion), the point clouds are transformed into virtual models (digital models) of the respective motifs. The motifs can be used for creating complex patterns. In this study, six complex patterns are created. The digital models are used to manufacture replicas of the respective motifs/patterns with an ordinary 3D printer. This way, this study digitally preserves the craftsmanship of Ainu motifs and patterns.

The proposed reverse engineering method equally applies to artifacts/objects other than Ainu motifs. It is demonstrated by producing a virtual model of an ancient ewer (3D shape). The processes involved in the proposed reverse engineering method (recursive point cloud creation process, point cloud coordination, solid CAD modeling, and 3D printing) do not require sophisticated devices. The results show that the proposed reverse engineering method is free from heavy computations, unlike conventional reverse engineering processes. As such, the outcomes of this study enrich the field of reverse engineering.

1.6. Thesis Structure

This thesis is divided into six chapters. The first presented the background, context, scope, aim and objectives, and contribution and significance of this study.

The second chapter addresses a literature review. The chapter presents reverse engineering, reverse engineering application in preserving cultural heritage, state-of-the-art 3D printing, and Hokkaido Ainu motifs and patterns.

The third chapter addresses the methodology of this study. The chapter presents the framework of the proposed reverse engineering method, recursive point cloud creation process to create an analytical point cloud, solid CAD modeling, 3D printing,

characteristics of the analytical point cloud, and quantification of the analytical point cloud.

The fourth chapter addresses the applications of the proposed reverse engineering method. The chapter presents two applications: the first application is a digitization and physical prototyping of ten selected basic Hokkaido Ainu and six complex patterns, and the second application is a digitization and physical prototyping of an ancient ewer.

The fifth chapter addresses the implications of this study and highlights the future direction of this study. The final chapter concludes this thesis.

Chapter 2: Literature Review

This chapter addresses comprehensive literature reviews related to this study. Section 2.1 presents literature reviews related to reverse engineering. The review includes an overview, definitions, different reverse engineering approaches, 3D scanning-based reverse engineering, and direction slicing of a scanned point cloud. Section 2.2 presents a comprehensive literature review on the application of reverse engineering in preserving cultural heritage. Section 2.3 presents a background related to 3D printing. The background covers an overview, benefits and limitations, basic steps of 3D printing, classifications of 3D printing, materials used, and applications of 3D printing. Section 2.4 presents a literature review on the Hokkaido Ainu motifs.

2.1. Reverse Engineering

The following subsections described a comprehensive literature review on reverse engineering and closely related issues.

2.1.1. Overview

Engineering is the branch of science and technology concerned with designing, building, and maintaining products, systems, and structures. There are two types of engineering at a higher level. Forward engineering and reverse engineering, as shown in Figure 1-1. Forward engineering is the conventional process of moving from high-level abstraction and logical designs to the physical implementation of a system. There may be a physical object without any technical details, such as drawings or documents. In such a case, the process of replicating existing parts in the form of a virtual model (digital model) is known as reverse engineering (RE).

In the past few decades, reverse engineering has brought a significant influence on the dissemination of technology. For example, reverse engineering enables the convergence of physical and digital worlds, taken advantage of by all the science and engineering fields. The first convergence began in the 1970s by digitizing sound using

signal processing that made analog to digital conversion of a common language. It helped the telecommunication industry and changed the ways of communication over a long distance. Such convergence of the physical and digital world was based on one-dimensional conversion. The second convergence began in the 1980s by digitizing fonts and pictures using image processing that made switching from paper documents to electronic documents. It helped the publishing industry and changed the way information is being stored and shared. Such convergence of the physical and digital was based on two-dimensional conversion. The third convergence began in the 1990s by digitizing the physical world to the digital world using geometric processing that made it possible for forward engineering to reverse engineering. It helped the manufacturing industry and changed the way the products were designed and manufactured. Such convergence of the physical and digital was based on three-dimensional conversion.

Reverse engineering covers object from a large aircraft to the smallest microchip, World War II's paranoia to the Cold War, commercial piracy to competitive intelligence, and product verification and analysis to crime preventive.

2.1.2. Definition

There are several definitions of reverse engineering. Some of the selected definitions are as follows. Rekoff [37] defined reverse engineering as "the process of developing a set of specifications for a complex hardware system by an orderly examination of specimens of that system." Chikofsky and Cross [38] defined reverse engineering as "the process of analyzing a subject system to identified the systems components and their relationships, and to create representations of the system in another form or at a higher level of abstraction." Kevin [39] defined reverse engineering as the process that "initiates the redesign process wherein a product is predicted, observed, disassembled, analyzed, test, 'experimented', and documented in terms of its functionality, form, physical principles, manufacturability, and assemblability." The IEEE Standard for Software Maintenance (IEEE Std 1219-1993) [40] defined reverse engineering as "the process of extracting software system information (including documentation) from source code." Society of Manufacturing Engineers (SME) [41] defined RE as "starting with a finished product or process and working backward in a logical fashion to discover the underlying new technology". The International Organization for Standardization (ISO)/

American Society for Testing and Material (ASTM) International [42] defined reverse engineering is as a "method of creating a digital representation from a physical object to define its shape, dimensions, and internal and external features".

2.1.3. Different Reverse Engineering Approaches

The state-of-the-art of reverse engineering is provided in multiple surveys conducted in [1], [13], [16], and [43]. In the literature, it was found that there are several approaches to reverse engineering. For example, when physical objects are available, then 3D scanning, photogrammetry (image processing), and sketching approaches were used. In the case of 3D scanning [4], an existing physical object is scanned by a 3D scanner and then represented by a relatively large set of points known as a point cloud. Afterward, the point cloud is processed by an appropriate computational method for making the point cloud meaningful to create a virtual model. In photogrammetry [5]/image processing [44], an existing physical object's image data are extracted. Afterward, the image data are processed using an appropriate computational method to make the image data meaningful to create a virtual model. In sketching [45-47], an existing physical object is sketched. The sketch data are then processed by an appropriate computational method for making the sketch data meaningful to create a virtual model.

On the other hand, some studies have shown entirely different approaches. Some selected approaches are reviewed as follows. Tamaki et al. [48] implemented a special type of photo-curing resin that can be used to extract the shape information of a physical object. For the sake of physical prototyping, they used a snow crystal. This method works well when the object is tiny. The parameters that affect the resin's curing process plays an essential role in the shape information extraction process. Rojas-Sola et al. [49] developed an alternative method. They used a 2D sketch of a historical artifact (in this case, an engine) as the reference model to construct the virtual model by using a commercially CAD system. This method relied on the model builder's perception. It worked well when the object consisted of simple geometrical shapes (e.g., plate, line, circle, cylinder, and similar). The authors did not report any results regarding the physical prototyping of the artifact (the engine). Furferi et al. [50] created a 2.5D tactical model from a painting (i.e., 3D arts), which helps individuals with vision problems experience the content of painting having historical significance. They took photographs of paintings and converted them to

solid CAD models based on the image processing method. Ullah et al. [33], [34], [35], [36], and [51] created point clouds by using a mathematical process such as iterative function systems and Monte Carlo simulation without using a 3D scanner. They used point clouds as base models and manufactured self-similar objects (e.g., fractals) and porous structures. However, this method is still in its infancy.

Among all the approaches above-mentioned, the 3D scanning approach is widely used in reverse engineering. Thus, the following review is narrowed to the 3D scanning-based reverse engineering.

2.1.4. 3D Scanning-based Reverse Engineering

As shown in Figure 1-1, reverse engineering consists of four steps: data acquisition, data preprocessing, surface reconstruction, 3D model creation, and physical prototyping. The issue pertaining to each step are described in the following subsections.

2.1.4.1. Data Acquisition

Data acquisition [7, 24] is the process of acquiring information using sensors or measurement devices. Different types of data acquisition methods are shown in Figure 2-1. As shown in Figure 2-1, a data acquisition method can be broadly classified into two methods: the contact method and the non-contact method. Both methods used some mechanism to interact with the surface of the physical object. In contact methods, the physical object's surface is touched by mechanical probes at the end of the arms. While in the case of non-contact methods, sound, light, or microwave are used. See in [7, 52] for more details.

3D scanning is one of the non-contact data acquisition methods widely used in reverse engineering. The ISO/ASTM [42] defined 3D scanning as "a method of acquiring the shape and size of an object as a 3-dimensional representation by recording x,y,z coordinates on the object's surface and through software the collection of points is converted into digital data".

In 3D scanning-based reverse engineering, a 3D scanner scans the targeted surface of the existing physical object, and it generates a relatively large number of points called point clouds. In a complex object, the 3D scanner cannot capture the entire targeted

surface from a single scan direction. It is required to scan multiple times by changing the orientation of the object. Each time the object is scanned and captures a portion of the total point cloud, referred to as a fragmented point cloud.

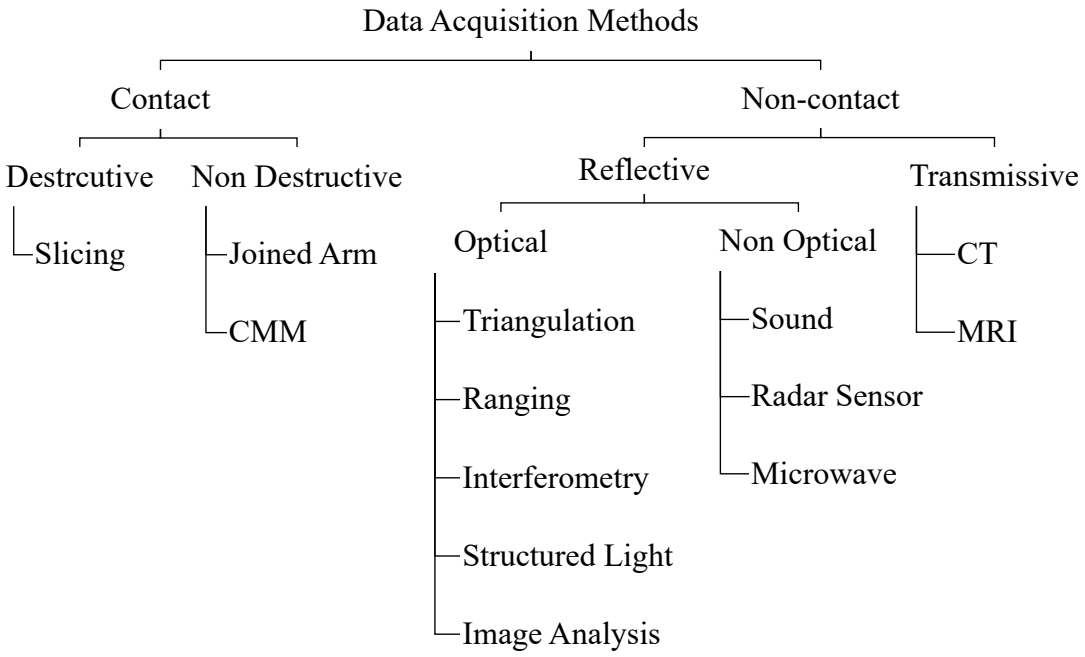


Figure 2-1. Classification of data acquisition methods [7, 52].

2.1.4.2. Preprocessing

Data preprocessing [25] is performed after completing the data acquisition step, as shown in Figure 1-1. The data preprocessing step consists of three procedures: registration, noise removal, and data reduction. The fragmented point clouds collected from multiple times are required to combine into a single coordinate system. This procedure is referred to as registration. The point cloud often contains noise and outlier points, and it is necessary to remove such unwanted points from the point cloud. The procedure of removing noise and outlier points from the point cloud is referred to as noise removable. The point cloud contains millions of points and requires to remove redundant point to save memory and increase computational speed. The procedure of removing the redundant points from the point cloud is referred to as data reduction.

Point cloud registration procedures are classified broadly into three categories: greedy searching-based, global feature-based, and local feature-based methods. See [53] for the details. Numerous researchers have developed different techniques or algorithms

to ensure accurate and efficient point cloud registration and noise removal. Some of the selected research are briefly discussed as follows. Li and Song [54] developed a modified iterative closest point algorithm based on the dynamic adjustment factor for registering point clouds. Zhao and Ilieş [55] implemented machine learning algorithms combined with the shape descriptor to classify 3D point clouds. Jung et al. [56] proposed an algorithm based on 2D synthetic images to improve point cloud registration rather than 3D point cloud data. James et al. [5] developed a photogrammetry color segmentation-based technique as an alternative approach to scan. They constructed 3D point clouds for reverse engineering applications. Yap et al. [57] implemented different data acquisition methods, such as engineering drawing, computed tomography, and optical coherence tomography, to build 3D printed bio-models for medical applications. They observed that the data acquisition and conversion technique had a significant effect on model quality. Syed et al. [31] investigated the influence of surface characteristics on point clouds. A point cloud is acquired using a laser scanning probe. They observed that the surface characteristics affect the quality regardless of the scanning accuracy.

In noise removal, common noise removal methods in reverse engineering are average, median, and Gaussian methods. All the methods are based on statistical approximation. Some researchers proposed a resampling approach. For example, Chen et al. [58] implemented centroidal Voronoi tessellation to resample point clouds to remove noise and fill holes or preserve point cloud boundaries. Skrodzki et al. [59] developed approaches to handle non-uniform density based on point cloud resampling.

On the other hand, in data reduction, the redundant points are removed from the flat regions and keep the points on highly sculptured regions such as corners, internal edges, and boundaries. Lee and Woo [60] developed an algorithm to remove redundant points from a point cloud to reduce the execution time to integrate reverse engineering and rapid prototyping.

2.1.4.3. Surface Reconstruction

Surface model reconstruction [26, 27] is performed after completing the data preprocessing step, as shown in Figure 1-1. Surface model reconstruction plays a critical role in reverse engineering. Compared to conventional surface reconstruction, reverse engineering puts a great deal of emphasis on freeform surfaces from the scanned point

clouds. The most commonly used geometrical modeling to convert a point cloud to a freeform surface are curve-based and polygon-based modeling [23]. The issues related to curve-based and polygon-based modeling are described in the following subsections.

(a) Curve-based Modeling

Curve-based modeling includes three procedures: segmentation, curve fitting, and surface fitting, as shown in Figure 1-1. The point cloud is first divided into subsets into a regular pattern using the segmentation process [28]. The cross-sections of the segmented point clouds are fitted using different parametric curves (NURBS, B-splines, and alike) [29] using the least-squares method. With the surface fitting or skinning function, the fitted curves are converted into a surface model.

Some of the selected studies are briefly discussed as follows. Le and Duan [61] proposed a segmentation algorithm for mechanical CAD models constructed from planes, cylinders, cones, spheres, and tori. Williams and Ilies [62] proposed a framework for performing shape analysis and segmentation directly on point clouds. Fayolle et al. [63] established a construction tree model for the surface model created from the point cloud. Pan et al. [64] implemented a phase-field guided surface reconstruction based on implicit hierarchical B-splines from the point cloud data. Feng and Taguchi [65] developed a T-spline fitting strategy algorithm for surface reconstruction from 2D point cloud data. Schwartz et al. [45] generated a 3D model by building a loft feature through some curves produced by slicing the point cloud. The point cloud data are obtained from 3D sketching. Xu et al. [66] implemented a virtual approach for slicing the point cloud for AM processing. Haiqiao et al. [67] developed algorithms for automatically generating blocked patterns from the point cloud data used in garment manufacturing. Peternell and Steiner [68] developed an algorithm for surface reconstruction of a piecewise planar object or CAD model from a point cloud. Pal [69] developed a tangent plane method for generating 3D geometry from a point cloud.

The currently available commercial curve-based modeling systems are Rhino 3D, Maya, Alias/Wavefront, CATIA, ICEM Surf, and SolidWorks. These systems can make it possible to create high-quality surface models. However, creating complex-shaped surfaces using these systems is still ill and challenging.

(b) Polygon-based Modeling

Polygon-based modeling is used as an alternative method to reconstruct a surface model from the point cloud. It consists of four procedures: triangulation, decimation, subdivision, and triangle to surface fitting. In triangulation, a polyhedral model is directly constructed by connecting neighboring points using one of the following methods: Delaunay triangulation, alpha shaping, crusting, or volumetric. In the decimation, the goal is to reduce the number of triangles in the polyhedral model, which can be achieved by edge collapse that calculates the cost of each edge by estimating the error that occurs when it is removed. In subdivision, it seeks to generate a smooth and fine surface from the coarse mesh. By applying a refined rule, a smooth and detailed model is obtained. Subdivision surface [70] schemes allow producing an approximated surface by adding vertices and subdividing existing polygons. Then, the surface is rendered on the triangulated model.

Some related work on curve-based modeling is discussed as follows. Masuda et al. [11] developed a method for reconstructing polygonal faces from the point cloud and mapped each point cloud onto a 2D image for detecting the bounded planar faces. Ma et al. [71] produced an umbrella facet matching algorithm to construct watertight manifold triangle meshes from a point cloud. Zhong et al. [72] developed an inverse distance square method for directly slicing spatial point cloud data from reverse engineering for rapid prototyping applications. Chang et al. [73] established a methodology for surface reconstruction from point clouds for creating textures on the surface that can be used in a medical scaffold. Remil et al. [74] proposed a framework to reconstruct 3D models from raw point cloud data by learning the prior knowledge of a specific class of objects.

The currently available commercial polygon-based modeling systems are MeshLab, PointCab, Kubit PointCloud for CAD, JRC 3D Reconstructor, Imagemodel, PolyWorks, Rapidform, Geomagic, Rhino 3D, Imageware, Geomagic, Rapidform, Paraform, and PolyWorks. Kuang-Hua [16] has provided a comparison of commercial modeling software for reverse engineering.

It is worth mentioning that compared to curve-based modeling, the polygon-based modeling process takes less time regardless of the geometric complexity of the scanned parts. However, it is challenging to reconstructing the fine details, such as the sharp

corners and edges, using the methods mentioned above. A comparison between curve-based modeling and polygon-based modeling are shown in Table 2-1.

Table 2-1. Comparison between curve-based and polygon-based modeling [23].

Parameters	Curve-based Modeling	Polygon-based Modeling
Steps	Segmentation, curve fitting, surface fitting	Triangulation, decimation, subdivision, surface fitting
Process	Manual	Automatic
Surface patching	Manual	Automatic
Surface quality	High	Medium or low
Features	Surface geometry only	Surface geometry and attributes (e.g., color)
Continuity	Cannot ensure continuity	Ensure continuity
Topology	Simple	Complex
Speed	A few weeks to a few months	A few hours to a few days

2.1.4.4. Virtual Model Creation

Virtual model creation is performed after completing the surface reconstruction step, as shown in Figure 1-1. The surface model reconstructed from the point clouds is an uneditable form. When the surface models are transferred to a 3D CAD system, the surface models are translated into an editable format known as a solid CAD model, which is a virtual model. Thus, the virtual model creation is also referred to as 3D CAD modeling or 3D CAD model creation [7, 13, 17, 75]. Reverse engineering can take advantage of the solid CAD model. For example, deriving new models, make variations, analyze properties, and determine characteristic quantities such as surface area and volume.

2.1.5. Direct Slicing of 3D Scanned Point Cloud

Some researchers deployed direct rapid prototyping from scanned point clouds without undergoing curve-based or polygon-based modeling to avoid complicated surface reconstruction.

Some related studies are discussed as follows. Percoco and Galantucci [76] developed a genetic algorithm for directly slicing point clouds for rapid prototyping. Wang et al. [77] implemented a multi-scale mesh-free empirical mode decomposition

(EMD) algorithm directly over a point cloud without the need for explicit mesh construction. Yang et al. [78] developed an algorithm based on the moving least-squared process. The process can intersect with lines, planes, polygonal mesh, and NURBS surfaces for directly manufacturing objects from point clouds without building a CAD model. Oropallo et al. [79] and [80] implemented a point cloud-based approach to directly slice a NURBS-based model for 3D printing rather than STL-based slicing.

2.2. Reverse Engineering in Preserving Cultural Heritage

With the advance in reverse engineering technology, reverse engineering has been used for digitizing, preserving, and restoring cultural heritage [32, 81-83]. For example, museums are using reverse engineering for making complete or missing parts of a vital relic [84, 85]. Even historical sites are digitally documented by using reverse engineering technology [86-88].

Like a 3D sculpture or architecture, a piece of 2D visual art such as a painting, photography, or drawing can be digitized in a solid CAD model (a virtual model) using reverse engineering. For instance, Fueferi et al. [50] have created a 2.5D tactical model from a painting (i.e., 2D art), which helps individuals with vision problems experience painting having historical significance. Thus, reverse engineering is an essential technology for preserving cultural heritage.

2.3. 3D Printing

Digital data of a virtual model created in reverse engineering are useful in producing a physical replica. The physical replica can be built using digital manufacturing processes, for example, additive manufacturing (e.g., 3D printing), subtractive manufacturing (e.g., CNC machining), or formative manufacturing (e.g., casting). The scope of manufacturing in this study is restricted to 3D printing. The following subsections described the background of 3D printing.

2.3.1. Overview

Contrary to the conventional manufacturing processes (e.g., subtractive manufacturing and formative manufacturing), 3D printing has been introduced to

manufacture physical objects by adding material layer by layer [22, 23]. 3D printing can manufacture physical objects having complex shapes and multi-scale structures. As a result, 3D printing is recognized as one of the vital technologies that will revolutionize the way products have been designed, manufactured, and marketed in the era of the 4th industrial revolution.

The ISO/ASTM [42] defined 3D printing as "the fabrication of objects through the deposition of a material using a printed head, nozzle, or another printer technology." 3D printing is commonly used as an alternative term for additive manufacturing (AM) [19-21]. The ISO/ASTM [42] defined AM as "a process of joining materials to make objects from 3D model data, usually layer upon layer, instead of subtractive manufacturing methodologies." Other terms used as a synonym of additive manufacturing include additive fabrication, freeform fabrication, rapid prototyping, desktop manufacturing, rapid manufacturing, and on-demand manufacturing.

The term 3D printing was coined in 1993 when MIT developed a powder bed process employing standard and custom inkjet print heads [89]. Later it was commercialized by Soligen Technologies, Extrude Hone Corporation, and Z Corporation. However, the foundations of 3D printing go back almost 150 years when freeform topographical maps and photo sculptures from 2D layers were built [90]. The concept of modern 3D printing started in the 1960s to the 1970s. In 1969, 1970, and 1979, the following 3D printing processes emerged: photopolymerization, powder bed fusion, and sheet lamination, respectively [75].

In 1981, the first successful 3D printing process was invented by Hideo Kodama of Nagoya Municipal Industrial Research Institute, Nagoya city, Japan. Kodama fabricated 3D plastic models using controlled ultraviolet light exposure on a photo-hardening [91] and the first person to the applied patent for rapid prototyping technologies.

In 1984, Charles Hull invented and patented a stereolithography apparatus [92] and contributed the STL file format, digital slicing, and infill strategies, commonly used in 3D printing. In 1989, Scott and Lisa Crump a Fused Deposition Modeling [20]. The current 3D printer in the market works based on Fused Deposition Modeling. 3D printing is now beginning to make significance in roadmaps [90, 93].

Numerous reputable market analysts [94, 95], industries [96, 97], and academics [98-100] has been continuously conducting research. They publish annual reports on the

current market trends, investment, and predicting the future of 3D printing. The International Data Corporation (IDC) [101] predicted that global spending on 3D printing would reach 13.8 billion dollars in 2019, increasing 21.2 percent over 2018. The IDC forecast that global spending is expected to reach nearly 22.7 billion dollars, with a five-year compound annual growth rate (CAGR) of 19.1 percent by 2020. According to the IDC, digital manufacturing dominated 3D printing more than half worldwide by spending throughout the 2018 – 2022 forecast, healthcare providers spending over $1.8 billion in 2019, and education spending a total of 1.2 billion dollars. The professional services are spending accounts for nearly 898 million dollars and customer spending almost 647 million dollars, which account for less than 5% of worldwide.

The 3D hubs [96] reviewed 3D printing trends 2020. They forecasted that the average CAGR at 24 percent for the next five years, reaching 34.9 billion dollars by 2024 [96]. The market value almost doubles every three years. According to the Gartner Institute report 2019 [94], the medical sector has grown in the use of 3D printing in 2018. By 2023, an expected 25 percent of medical devices will make use of 3D printing. In 2019, metal 3D printing was on the rise. By 2020, parts built using metals and allows will become a critical element in supply chains for replacement parts both in commercial and military applications. In brief, the 3D printing market share is exponentially increasing.

2.3.2. Benefits and Limitations

There are several benefits of 3D printing. Some of the benefits are the freedom to design highly customized products, the realization of complex geometry, the fabrication of multi-material objects, minimal material waste, complexity for free, and rapid prototyping [19, 102]. However, there are also some limitations of 3D printing. The limitations are slow to build rates, high production costs, the considerable effort required for application design, discontinuous production process, and limited component size.

2.3.3. Basic Steps

A typical 3D printing consists of the following basic five steps, as schematically illustrated in Figure 2-2 shows [103].

1. 3D model creation of the desired object

2. Triangulation of the 3D model.
3. Slicing of the STL file format into 2D cross-sectional layers.
4. Execute the 3D printing process.
5. Postprocessing.

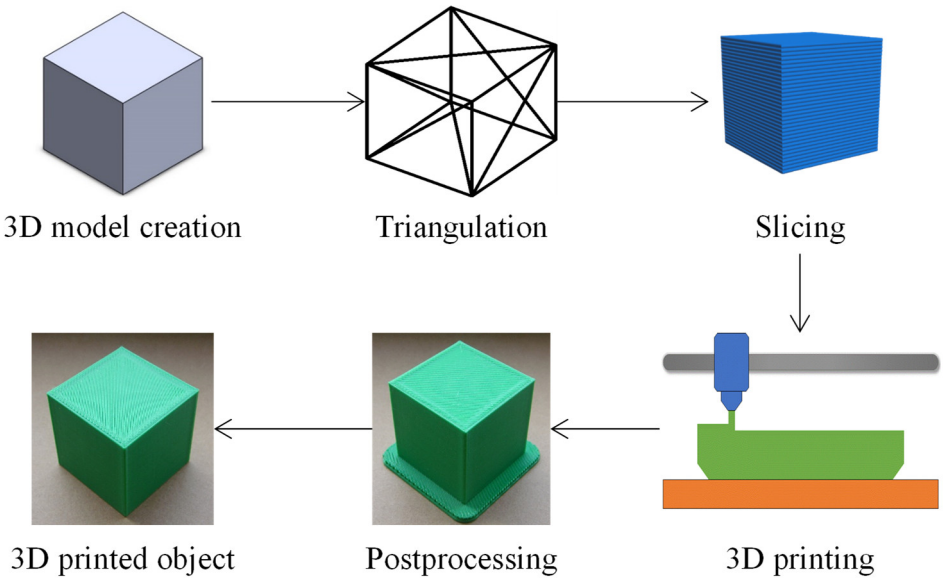


Figure 2-2. Typical 3D printing steps.

In the first step, A 3D model of the desired object is created using an off-the-shelf 3D CAD system or other means. The 3D models can be built based on either concept designs, importing existing CAD files, or virtual models derived from reverse engineering. The 3D model is also referred to as a solid CAD model [29]. Some of the commercial CAD systems available in the market for designing objects for 3D printing are AutoCAD, Fusion360, Onshape, Pro/Engineer, Rhinoceros, SolidWorks, and ThinkCAD.

In the second step, triangulation is performed. Triangulation is a process that transforms a solid CAD model into a triangulated model. The triangulated model represents a 3D surface or an outer boundary of the solid CAD model as planar triangles, resulting in facets. The triangulated model dataset is referred to as standard triangle language (STL) data that contains the vertices and direction of each triangle's outward normal. For a simple model such as a cube box shown in Figure 2-2, its surface can be approximated with twelve triangles. For the complex surfaces, a few triangles to several thousand triangles are used. The STL file format has been used as the de facto standard in the rapid prototyping industry, including 3D printing [104]. The STL file format has

been the best suitable file format for slicing the triangulated model. Nowadays, 3D CAD systems are capable of generating an STL file format. The user can convert the CAD file into the STL file format by merely saving files as STL. The limitations of the STL data are data size larger than the original CAD data, the geometry flaws may exist, and slicing of large STL data can take long hours. Some of the software packages available in the market for modifying or repairing STL are Meshmixer and Netfabb.

In the third step, an STL file format is sliced into 2D cross-sectional layers. Slicing divides the triangulated model into many thin horizontal layers. It then generates a set of instructions in machine language (g-code) for the 3D printer to execute. Slicing is performed using software packages. In other words, the slicing process converts the STL file format into a g-code file format that provides the paths to the printhead of a 3D printing device. The 2D cross-sectional layer thickness may vary from 0.01 mm to 0.7 mm, depending on the 3D printing device. The thinner the layer increases the accuracy but also increase the build time. The slicing process also helps to generate support for the model during its buildup. Nowadays, the dedicated slicing software comes together with the 3D printer. Some of the slicing software packages available in the market are ideaMaker, Cura, Slic3r Simplify3D.

In the fourth step, execute the 3D printing process. The 3D printing process starts with uploading g-code data into a 3D printer, selecting desired printing materials, and running the 3D printing devices (e.g., 3D printer). It may take a few minutes to a couple of hours, depending on the object's size, to be built.

The final step is postprocessing. The preprocessing includes removing the 3D printed object from the device, cleaning, removing support materials, and surface finishing, such as sanding, sealing, or painting to improve mechanical properties, accuracy, and appearance. See the design for additive manufacturing in [105].

2.3.4. Classification

The ISO/ASTM 52900 standard [42] has classified additive manufacturing (3D printing) processes into seven categories, as listed below, and their basic principle and technologies are summarized in Table 2-2.

- Binder jetting (BJ)
- Directed energy deposition (DEP)

- Material extrusion (ME)
- Material jetting (MJ)
- Powder bed fusion (PBF)
- Sheet lamination (SL)
- Vat photopolymerization (VP)

Table 2-2. Summary of the 3D printing processes [21, 106].

Processes	Basic Principle	Technology
Binder jetting (BJ)	Liquid binder applied onto thin layers of powder material.	- 3D inkjet
Directed energy deposition (DED)	Fused thermal energy melts fuse materials during deposition.	- Laser metal deposition (LMD) - Laser engineered net shaping (LENS) - Direct metal deposition (DM3D) - Electronic beam additive manufacturing (EBAM)
Material extrusion (ME)	Build material is extruded through a nozzle or orifice.	- Fuse deposition modeling (FDM) - Fuse filament fabrication (FFF)
Material jetting (MJ)	Droplets of build materials are selectively deposited layer by layer to build parts.	- Poly jet - Smooth curvatures printing (SCP) - Multi-jet modeling (MJM)
Powder bed fusion (PBF)	Thermal energy selectively melts regions of a powder bed of the build material.	- Selective laser sintering (SLS) - Direct metal laser sintering (DMLS) - Selective laser melting (SLM) - Electron beam melting (EBM) - Selective heat sintering (SHS) - Multi-jet fusion (MJF)
Sheet lamination (SL)	Sheets or foils of material are bonded to form an object.	- Ultrasonic additive manufacturing (UAM) - Selective deposition lamination (SDL) - Laminated object manufacturing (LOM)
Vat photopolymerization (VP)	Liquid photopolymer in a vat is selectively cured by light-activated polymerization. It converts the exposed areas to a solid part.	- Stereolithography (SLA) - Direct light processing (DLP) - Continuous liquid interface production (CLIP) - Scan, spin, and selectively photocure (3SP)

2.3.5. Materials

3D printing uses the following types of materials: polymers, metals, ceramics, composites, concrete, papers, and certain edibles (e.g., chocolate). The polymer materials include acrylonitrile butadiene styrene (ABS), acrylonitrile styrene acrylate (ASA), cellulose, nylon, resin, polylactic acid (PLA), polyethyleneimine (PEI), and thermoplastic polyurethane (TUP). The metal material includes titanium (Ti), Tantalum (Ta), copper (Cu), gold (Au), and silver (Ag), and allow metal materials include Ti-based, Ni-based, Fe-based, Al-based, cobalt-based, and Cu-based. Materials often come in powder form, liquid resin, or wire filament.

2.3.6. Applications

Nowadays, 3D printed objects are seen everywhere. 3D printing is deployed in numerous industrial sectors, including the automotive industry [107], the aerospace industry [75], the biomedical engineering field (e.g., dental braces, artificial tissues, hearing aids, dummy cadavers, and prosthetics) [9, 108-113], the education field [114-116], the fashion industry (e.g., cloth, shoes, and jewelry) [117-121], the food industry (e.g., pizza, chocolate, crackers, and pasta) [122-126], and the construction industry (e.g., 3D printed houses) [127-130]. 3D printing has also been applied to preserving and restoring cultural heritage [81]. Small, affordable, and user-friendly 3D printing machines (e.g., 3D printers) are now available to running small businesses [131-133].

2.4. Hokkaido Ainu Motifs and Patterns

This section describes the Hokkaido Ainu motifs and patterns. The Ainu are indigenous people living in the northern part of the Japanese archipelago, namely Hokkaido, Aomori, and Sakhalin peninsula [134, 135]. The Ainu decorate their houses, clothing, ornaments, utensils, and spiritual goods using unique or exotic patterns [136-140].

This review is focused on the motifs and patterns belonging to the Ainu community living in Hokkaido. The Ainu communities living in other regions have similar patterns but different meanings that vary from region to region.

The patterns carry their identity and a sense of aesthetic. Notably, the Hokkaido Ainu believe that wearing patterns on their cloth protects them against evil, danger, or disease. The patterns have vital cultural significance. Nowadays, different kinds of souvenirs, shirts, and bags are crafted with Hokkaido Ainu patterns cherished by many individuals in Japan and abroad (Ref. [26]). There are even shopping streets (e.g., the shopping street in Akan Sap [141]) specialized in products crafted with the Ainu motifs. Goods crafted with patterns have vital commercial significances to the Hokkaido Ainu. Figure 2-3 shows some glimpses of Hokkaido Ainu patterns crafted on houses and clothing taken recently at the shopping street in Akan, Kushiro, and the Hokkaido Museum of Northern Peoples, Abashiri.

(*a*) Hokkaido Ainu patterns embroidered on cloths

(*b*) Hokkaido Ainu patterns are crafted on house

(*c*) Akan street, Kushiro

Figure 2-3. Some glimpses of Hokkaido Ainu patterns.

The patterns on the clothing are made by patchwork and embroidery, and the skill of the craft is usually passed from mother to daughter. The mathematical representations of the basic motifs and the underlying patterns are unavailable. A great deal of craftsmanship is needed to create such patterns. The remarkable point is that there is a lack of human resources to create patterns. This trend will remain the same in the foreseeable future, necessitating a pressing need for preserving craftsmanship. One of the pragmatic options for preserving the craftsmanship is to use digital manufacturing technology called reverse engineering. From this perspective, this study focused on modeling the Hokkaido Ainu patterns.

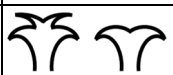
Numerous authors have reported the patterns (e.g., see the work [136-138]) used by the Hokkaido Ainu community [139]. The patterns are created by synthesizing basic motifs. The Sapporo city authority has summarized the basic motifs into fourteen types, which are listed in Table 2-3.

As seen in Table 2-3, motif Number 1 takes the shape of intersects each other is called Utasa in the Ainu language. Motif Number 2 takes the shape of a spiral is called Morew in the Ainu language. Motif Number 3 takes the shape of an eye is called Sik in the Ainu language. Motif Number 4 takes the shape of a thorn is called Ayus in the Ainu language. Motif Number 5 takes the shape of two spirals is called Uren-morew in the Ainu language. Motif Number 6 takes the shape of two spiral shapes with an eye called Ski-uren-morew in the Ainu language. Motif Number 7 takes the shape of a spiral with a small thorn is called Arus-morew in the Ainu language. Motif Number 8 takes the shape of a spiral with corners and edges is called Sikikew-nu-morew in the Ainu language. Motif Number 9 takes the shape of a spiral plant is called Morew-etok in the Ainu language. Motif Number 10 takes the shape of a vane is called Punkar in the Ainu language. Motif Number 11 takes the shape of a flower is called Apapo-piras (u) ke in the Ainu language. Motif Number 12 takes the shape of a flower bud is called Apapo-epuy in the Ainu language. Motifs Number 13 and 14 shapes do not have Ainu names, but they look like a heart-type shape and a temple bell shape, respectively.

The motifs called Utasa, Morew, Sik, and Ayus are frequently used and classified as the main motifs. Uren-morew, Ski-uren-morew, Arus-morew, and Sikikew-nu-morew are the motifs created by modifying the main motifs and classified as synthetic motifs. The motifs called Morew-etok, Punkar, Apapo-piras (u) ke, and Apapo-epuy represent

plants and classified as plant motifs. The other motifs (i.e., motifs Number 13 and 14) can be classified as other motifs.

Table 2-3. Basic motifs used by the Ainu community living in Hokkaido [142].

Number	Motif	Ainu Name	Description	Classification
1		Utasa	The motif crosses each other	Main motifs
2		Morew	The motif looks like a whirling or spiral	
3		Sik	The motif looks like eyes	
4		Ayus	The motif looks like (God's) thorn	
5		Uren-morew	The motif has two whirls	Synthetic motifs
6		Ski-uren-morew	The motif has two whirls that look like eyes	
7		Arus-morew	Whirling motifs has small thorns	
8		Sikikew-nu-morew	Whirling motif has corners and edges	
9		Morew-etok	The motif looks like a spiral plant	Plant motifs
10		Punkar	The motif looks like a vine	
11		Apapo-piras (u) ke	A flower in bloom motif	
12		Apapo-epuy	A flower bud motif	
13		---	A heart-type motif	Other motifs
14		---	Temple bell-type motif	

This chapter represented a comprehensive literature review on reverse engineering, the application of reverse engineering for preserving cultural heritage, 3D printing, and Hokkaido Ainu motifs. The next chapter presents the methodology of this study.

Chapter 3: Methodology

This chapter addresses the methodology of the proposed novel reverse engineering method suitable for 3D printing and its application for preserving cultural heritage. The chapter presents the framework of the proposed reverse engineering method (Section 3.1), recursive process (Section 3.2), point cloud coordination (Section 3.3), solid CAD modeling (Section3.4), and 3D printing (Section 3.5). The chapter also presents the characteristics of analytical point clouds (Section 3.6) and the quantification of analytical point clouds (Section 3.7).

3.1. The Framework of the Proposed Method

This section describes the framework of the proposed reverse engineering method. In the proposed reverse engineering method, four processes were considered, namely, recursive point cloud creation process, point cloud coordination, solid CAD modeling, and 3D printing. Figure 3-1 schematically illustrates the framework of the proposed reverse engineering method.

As shown in Figure 3-1, the process begun with the recursive point cloud creation process. In this process, first, study the given object and determine the features to be modeled. Then, fixed the values of the input parameters of the recursive point cloud creation process, namely, center point P_c, initial length d, initial angle ϕ, instantaneous distance $(r_i|i = 0,1,\ldots,n)$, and instantaneous rotational angle $(\theta_i|i = 0,1,...,n)$. Once the input settings were satisfied, the recursive point cloud creation process was executed and collected the output points denoted as an elementary point cloud. The recursive point cloud creation process was repeated until all the required elementary point clouds of the given object's features were created. In the point cloud coordination, the elementary point clouds were joined according to the order of features. In solid CAD modeling, the coordinated point clouds were exported to an appropriate 3D CAD system. Then, solid CAD modeling was performed using the 3D CAD system's function, which results in a solid CAD model (i.e., virtual or digital). Finally, in the 3D printing process, a physical object was built from the triangulated data (often referred to as STL data) of the solid

CAD model using a commercially available 3D printer. Other manufacturing means can be used if preferred.

In this study, the term a physical model, replica, and 3D printed object are interchangeable. Also, the term a virtual model, digital model, and solid CAD model are interchangeable. The next section further elaborated on the processes of the proposed reverse engineering method.

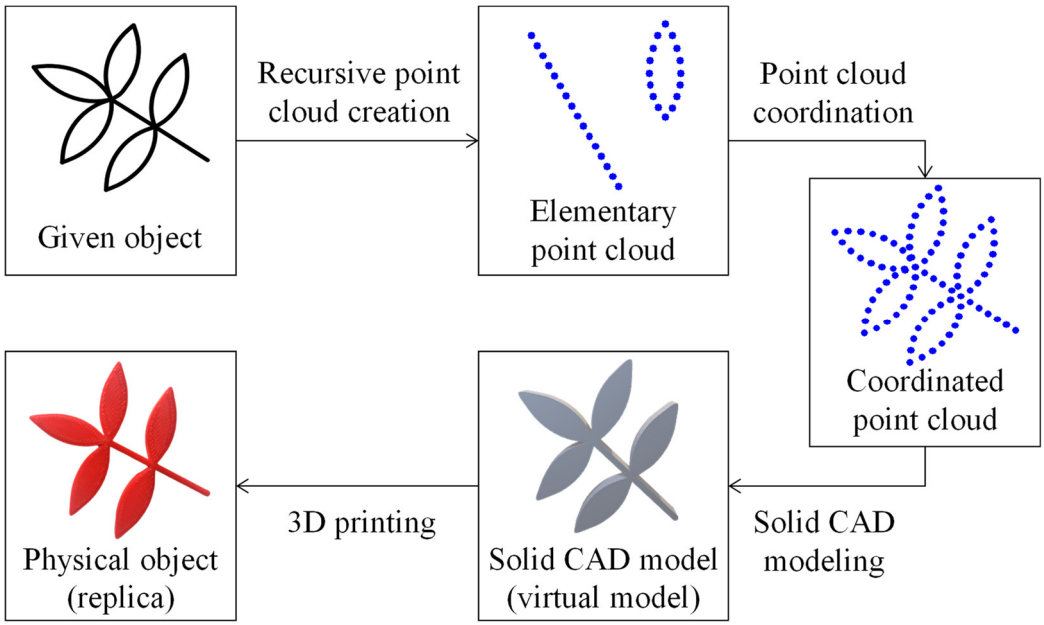


Figure 3-1. The framework of the proposed reverse engineering method.

3.2. Recursive Point Cloud Creation Process

This section described a recurve point cloud creation process (hereafter referred to as a recursive process) to create an analytical point cloud. Ullah et al. [33] have shown the recursive process. Setpoints were created recursively by two user-defined parameters called distance and angle. We had modified the recursive process to create an analytical point cloud. Algorithm-1 shows the steps to create an analytical point cloud using the recursive process.

As shown in Algorithm-1, the recursive process consists of four steps. The first step is the input step, the second step is the calculation step, the third step is the iteration step, and the last step is the output step. In the input step, the center point $P_c = (P_{cx}, P_{cy}) \in \Re^2$, initial distance $d > 0$, initial angle $\phi \in \Re$, instantaneous distances ($r_i \in \Re \mid i = 0,1, ..., n$), and instantaneous rotational angles ($\theta_i \in \Re \mid i = 0,1, ..., n$) are defined. In the calculation

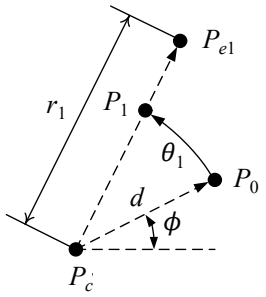
step, the initial point $P_0 = (P_{0x}, P_{0y})$ is calculated, which is a point at a distance d from P_c on the line P_cP_0, making an angle ϕ with the x-axis in the counter-clockwise direction. In the iteration step, P_0 is first rotated in the counter-clockwise direction of the x-axis using θ_i to create the points $P_i = (P_{ix}, P_{iy})$, $i = 0,1, \ldots, n$. Afterward, the points denoted as $P_i = (P_{ix}, P_{iy})$, $i = 0,1, \ldots, n$, are placed at the distances given by r_i from P_c resulting in the points $P_{ei} = (P_{ix}, P_{iy})$, $i = 0,1, ..., n$. In the output step, the points $P_{ei} = (P_{ix}, P_{iy})$, $i = 0,1, ..., n$, are collected as the point cloud denoted as $PC = \{P_{ei} \mid i = 0,1, ..., n\}$.

Algorithm-1: Recursive Process		
1	Define:	Center point $P_c = (P_{cx}, P_{cy}) \in \Re^2$ Initial length $d > 0$ Initial angle $\phi \in \Re$ Instantaneous distance $r_i \in \Re \mid i = 0,1, ..., n$ Instantaneous rotational angle $\theta_i \in \Re \mid i = 0,1, ..., n$
2	Calculate:	$P_0 = (P_{0x}, P_{0y})$ so that $P_{0x} = P_{cx} + d\cos\phi$ $P_{0y} = P_{cy} + d\sin\phi$
3	Iterate:	For $i = 0,1, \ldots, n$ Rotate P_0 by an angle θ_i around P_c in the counter-clockwise direction to create $P_i = (P_{ix}, P_{iy})$ so that $P_{ix} = P_{cx} + (P_{0x} - P_{0x})\cos\theta_i - (P_{0x} - P_{0x})\sin\theta_i$ $P_{iy} = P_{cy} + (P_{0y} - P_{0y})\sin\theta_i + (P_{0y} - P_{0y})\cos\theta_i$ Extend P_i to P_{ei} (a point on the line P_cP_i at a distance r_i from P_c) $P_{eix} = P_{cx} + (P_{ix} - P_{cx})\,(r_i/d)$ $P_{eiy} = P_{cy} + (P_{iy} - P_{cy})\,(r_i/d)$ End for
4	Output:	Point cloud $PC = \{P_{ei} \mid i = 0,1, \ldots, n\}$

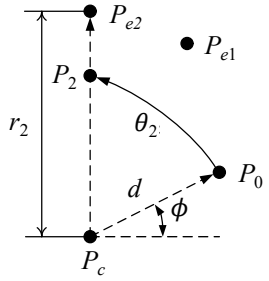
The working principle of the recursive process is schematically illustrated in Figure 3-2. The first four iterations for some arbitrary values of the instantaneous distance and instantaneous rotational angle.

As seen in Figure 3-2, the recursive process creates points with respect to the center point P_c and initial point P_0 depending on the value of the instantaneous distances r_i and instantaneous rotational angles θ_i. The recursive process can be used to create different

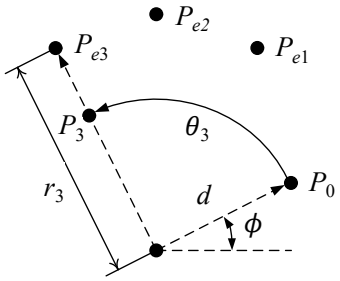
kinds of planer shape. In this respect, the user needs to choose the right pairs of instantaneous distances and instantaneous rotational angles. In most cases, a simple monotonic function (a straight line with positive or negative slope) or tent-like function can be used to fix the values of instantaneous distances and rotational angles.



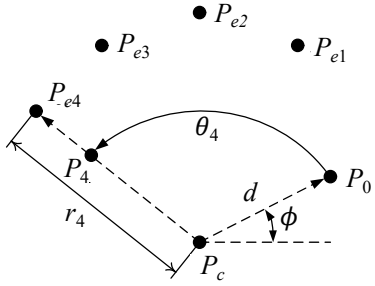
(*a*) The first iteration



(*b*) The second iteration



(*c*) The third iteration



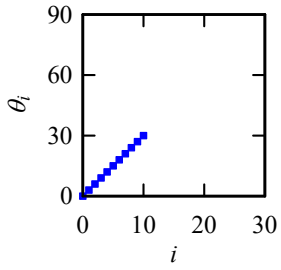
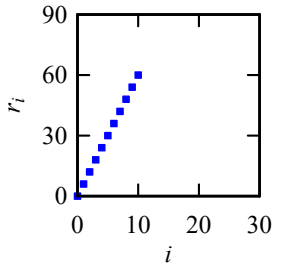
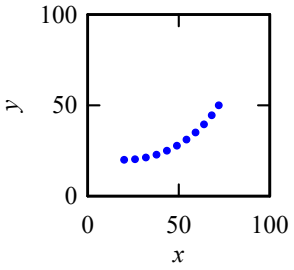
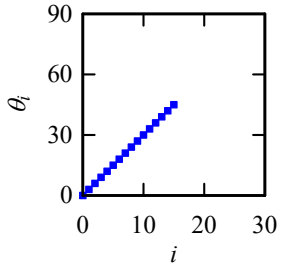
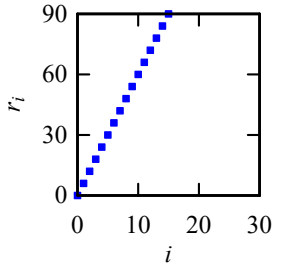
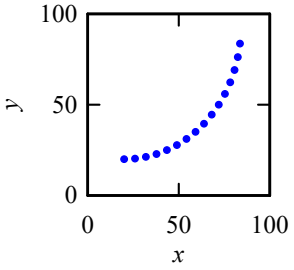
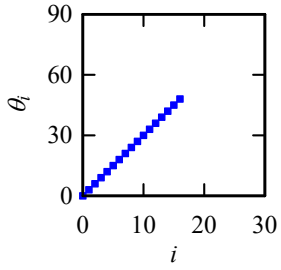
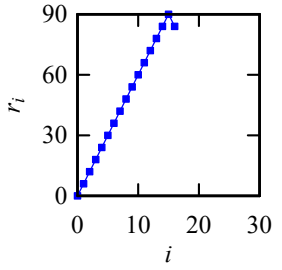
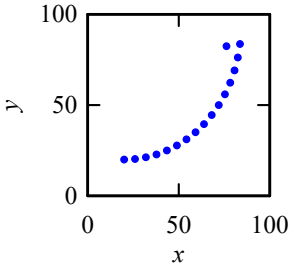
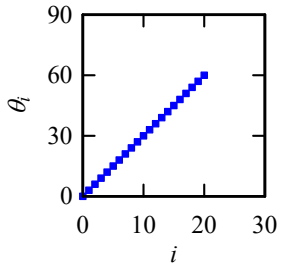
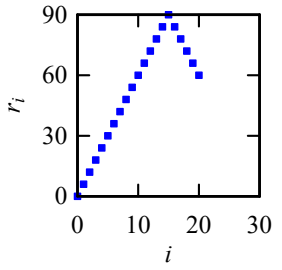
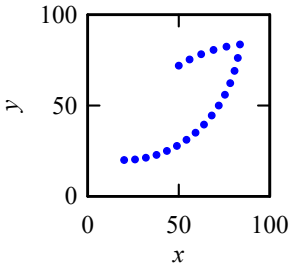
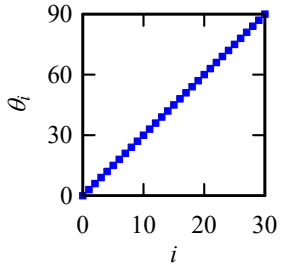
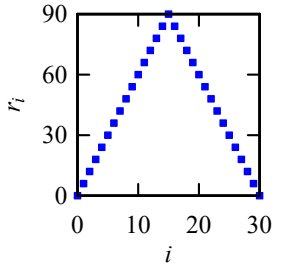
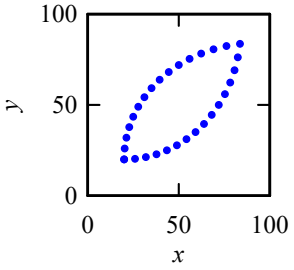
(*d*) The fourth iteration

Figure 3-2. The working principle of the recursive process.

Table 3-1 shows an example of how a point cloud is evolved when the instantaneous rotational angles are varied in straight-line and the instantaneous distances were varied in tent-function. This example reveals that at the point of inflection (see the plot of r_i), the point cloud changes its direction. Simultaneously, the other parameter keeps its monotonicity (compare the plot of θ_i with r_i).

As seen in Table 3-1 that the instantaneous rotational angle and instantaneous distance are instrumental parameters. Therefore, when the recursive process models a representative segment of a shape, the user must know how to fix the value of instantaneous distance and instantaneous rotational angle, further elaborated later after describing the proposed reverse engineering method processes.

Table 3-1. Gradual creation of a point cloud by the recursive process.

Iteration (i)	Inputs		Output
	Instantaneous Rotational angle (θ_i)	Instantaneous Distance (r_i)	Point Cloud (PC)
10			
15			
16			
20			
30			

3.3. Point Cloud Coordination

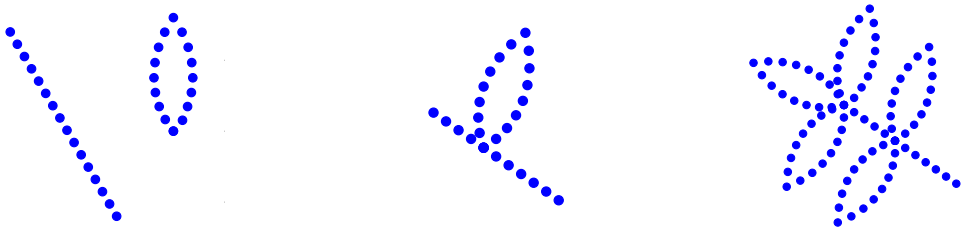
According to the given object's features' order, the elementary point clouds created in the recursive process must be coordinated or synthesized. The elementary point clouds were coordinated using an algorithm denoted as the point cloud coordination algorithm (PCCA). The mathematical setting of the PCCA is shown in Algorithm-2.

Algorithm-2: Point Cloud Coordination Algorithm (PCCA)		
1	Define:	The point cloud vector $PCV = (PC_1, PC_2, \ldots, PC_n)$, where $PC_j = \{P_{ejk} = (P_{exjk}, P_{eyjk}) \mid \forall j \in \{1,2, \ldots, n\}, k = 1,2, \ldots, n\}$
2	Reference:	Select PC_j as a reference point cloud
3	Iterate:	For $j = 1, \ldots, n$ Select the knots from two adjacent point clouds (j-th point cloud and (j+1)-sh point cloud) $P_{nt(j)} = (P_{ntx(j)}, P_{nty(j)}) \in PC_j$ $P_{nt(j+1)} = (P_{ntx(j+1)}, P_{nty(j+1)}) \in PC_{(j+1)}$ Calculate $dx = P_{ntx(j)} - P_{ntx(j+1)}$ $dy = P_{nty(j)} - P_{nty(j+1)}$ Update (j+1)-sh point cloud For $k = 1, \ldots, n$ $P'_{ex(j+1)} = P_{ex(j+1)k} + dx$ $P'_{ey(j+1)} = P_{ey(j+1)k} + dy$ $P'_{e(j+1)} = (P'_{ex(j+1)}, P'_{ey(j+1)})$ End for End for
4	Output:	Point cloud, $PC_{out} = \{PC_j, P'_{e(j+1)} \mid j = 1, \ldots, n\}$

The PCCA consists of four steps. The first is the input step, the second is the reference step, the third is the iteration step, and the last is the output step. In the input step, the point cloud vector $PCV = (PC_1, PC_2, \ldots, PC_n)$, where $PC_j = \{P_{ejk} = (P_{exjk}, P_{eyjk}) \mid \forall j \in \{1,2, \ldots, n\}, k = 1,2, \ldots, n\}$ is defined. In the reference step, the point cloud PC_j is selected as reference point cloud. In the iteration step, first, select the knots from two adjacent point clouds (j-th point cloud and (j+1)-sh point cloud). Then, calculate the difference between two knots as $dx = P_{ntx(j)} - P_{ntx(j+1)}$ and $dy = P_{nty(j)} - P_{nty(j+1)}$. Next, update

(j+1)-sh point cloud. Then, determine the final point cloud as $P'_{ex(j+1)} = P_{ex(j+1)k} + dx$ and $P'_{ey(j+1)} = P_{ey(j+1)k} + dy$ that results as $P'_{e(j+1)} = (P'_{ex(j+1)}, P'_{ey(j+1)})$. In the output step, the points $P'_{e(j+1)} = (P'_{ex(j+1)}, P'_{ey(j+1)})$, $i = 0,1, ..., n$, $k = 1, \ldots, n$ is collected as the point cloud denoted as $PC_{out} = \{PC_j, P'_{e(j+1)} | j = 1, \ldots, n\}$. The output point cloud maintains a C^0 continuity.

Besides the coordinating, other operations such as scaling, translation, and rotation can be performed in the point cloud coordination, if needed. In some cases, a single point cloud was enough to model the given object. In such a case, the point cloud coordination process is not required. Figure 3-3 shows the example of point cloud coordination using the PCCA. The settings to create a point cloud is shown in Table 3-1.



(a) Elementary point clouds of a line and leaf-shape

(b) Point clouds of the line and leaf-shape are joined using the PCCA

(c) Desired point cloud is created by copying and rotating the leaf-shape

Figure 3-3. Example of the point cloud coordination.

3.4. Solid CAD Modeling

In solid CAD modeling, a solid CAD model was constructed from the point cloud. The steps of solid CAD modeling are schematically illustrated in Figure 3-4. Solid CAD modeling was performed in SolidWorks (a 3D CAD software from Dassault Systems) [143].

As shown in Figure 3-4, first, the point clouds were exported to a 3D CAD system, then the applied entity, and created offsets using features available in the 3D CAD system. In some cases, the offset creations were not necessary. Finally, the selected surfaces and then applied extrusion, which results in a solid CAD model. Other functions such as lofts, revolve, sweeps, or thicken can be applied based on the requirements. Detailed steps of point cloud-based solid CAD modeling in SolidWorks are shown in Appendix A.

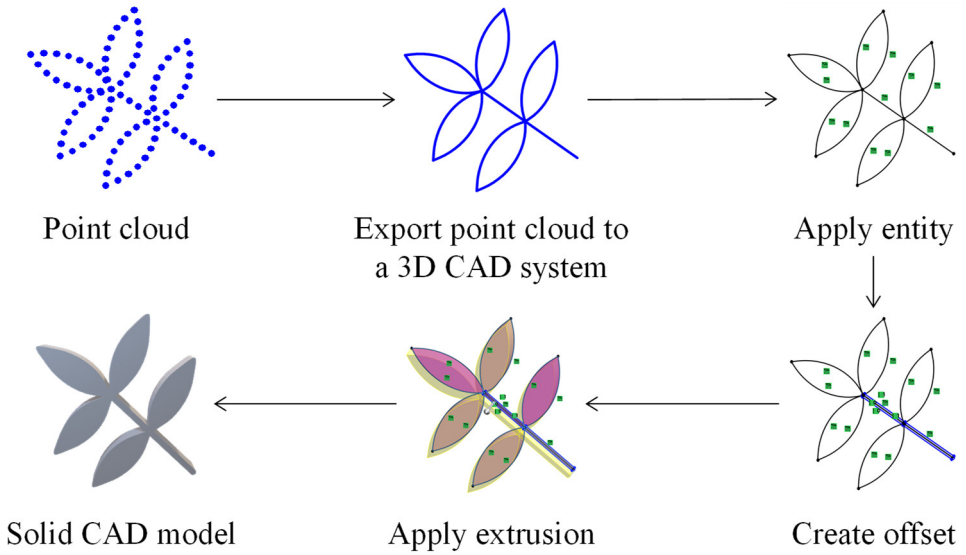


Figure 3-4. Steps of solid CAD modeling.

3.5. 3D Printing

In 3D printing, the solid CAD model is transformed into a physical model. The steps of the 3D printing process are shown in Figure 3-5.

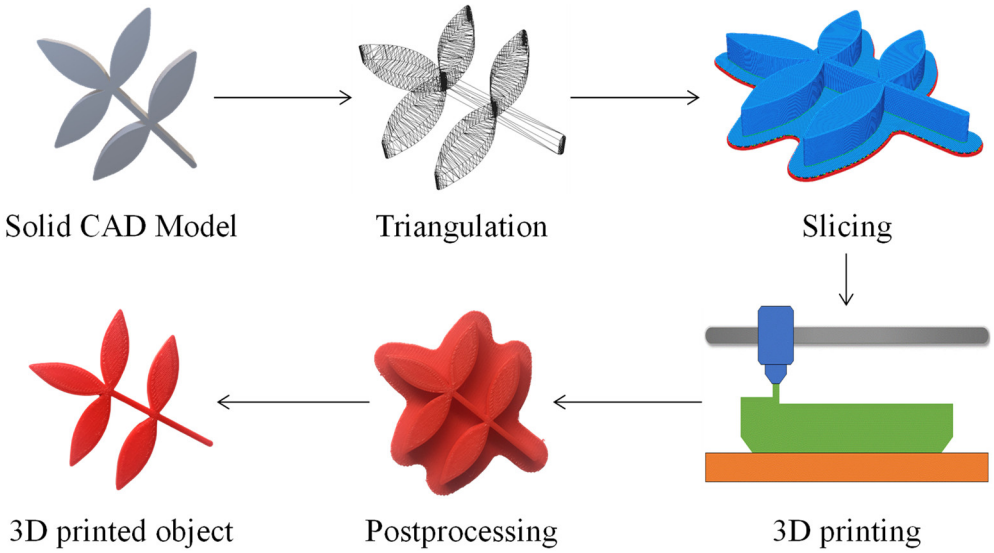


Figure 3-5. Steps of 3D printing.

As shown in Figure 3-5, the solid CAD model was transformed into a triangulated model using the 3D CAD system by saving the solid CAD model as an STL file format. Detailed steps of triangulation in the SolidWorks are shown in Appendix B. Then, the STL data was uploaded in slicing software. The slicing software sliced the triangulated

model into layers and generated the g-code. Afterward, the g-code was uploaded into a 3D printer, selected print material, and executed the 3D printing process to build the physical model. The 3D printing process may take a few minutes to a couple of hours, depending on the object's build size. After completing the 3D printing process, the 3D printed object was removed from the 3D printer, and the postprocessing is performed. In the postprocessing, the support materials were removed from the 3D printed object.

In this study, the physical objects were built using a 3D printer made by the Raise3D (model Pro2) [144], available at the Kitami Institute of Technology. The ideaMaker software package (slicing software) comes together with the 3Dprinter is used for the slicing process. The 3D printed object models were built using polylactic acid (PLA) materials.

3.6. Characteristics of Analytical Point Cloud

This section described the characteristics of an analytically point cloud created by the recursive process. As shown in Algorithm-1, the recursive process needs three constant inputs and two variable inputs. Three constants inputs are the center point P_c, initial distance d, and initial angle ϕ, which remain constant for all iterations. Two variable inputs are instantaneous rotational angle θ_i and instantaneous distance r_i, which varies in each iteration. The output of the recursive process is a point cloud, denoted as $PC = \{P_{ei} \mid i = 0, 1, ..., n\}$. The PC is a small-size point cloud, and its characteristics are depended on input parameters.

The characteristics of the analytical point cloud must be known beforehand to facilitate the modeling process. Accordingly, the analytical point cloud characteristics created by the recursive process are elucidated using some well-known shapes. For example, straight-line, circle, ellipse, spiral, astroid, S-shape, and leaf-shape are considered. The description is as follows.

Let consider first the point clouds that evolve by setting the instantaneous rotational angle and instantaneous distance linearly. These settings create some point clouds representing a straight line, circle, and spiral, as shown in Figure 3-6 to Figure 3-9.

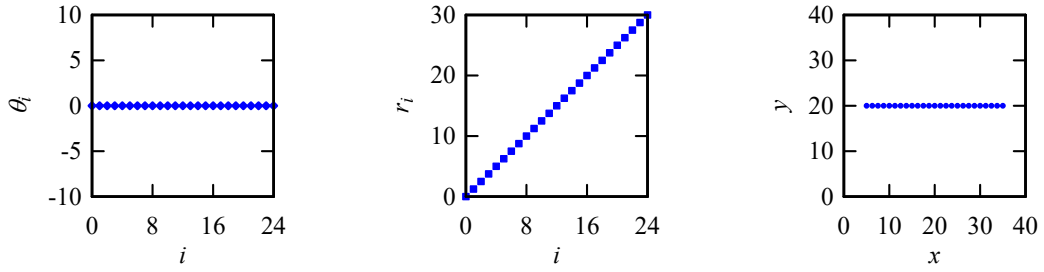
For example, consider a point cloud that models a straight-line. In this case, the instantaneous rotational angle θ_i must be kept constant, and the instantaneous distance r_i

must linearly increase or decrease. For instance, when the instantaneous rotational angle $\theta_i = 0°$, and the instantaneous distance r_i linearly increases for all iterations. The recursive process creates a point cloud *PC* representing a horizontal line, as shown in Figure 3-6. Similarly, when the instantaneous rotational angle $\theta_i = 90°$, and the instantaneous distance r_i linearly increases for all iterations. The recursive process creates a point cloud *PC* representing a vertical line, as shown in Figure 3-7. The slope of the straight-line can be modeled using an instantaneous rotational angle θ_i.

When the opposite strategy is adopted, the instantaneous rotational angle θ_i increases linearly, and the instantaneous distance r_i is kept constant. The recursive process creates a point cloud *PC* representing a circle, as shown in Figure 3-8. The radius of the circle can be modeled using instantaneous distance r_i. On the other hand, when both the instantaneous rotational angle θ_i and instantaneous distance r_i are increased linearly, the recursive process creates a point cloud *PC* representing a spiral, as shown in Figure 3-9.

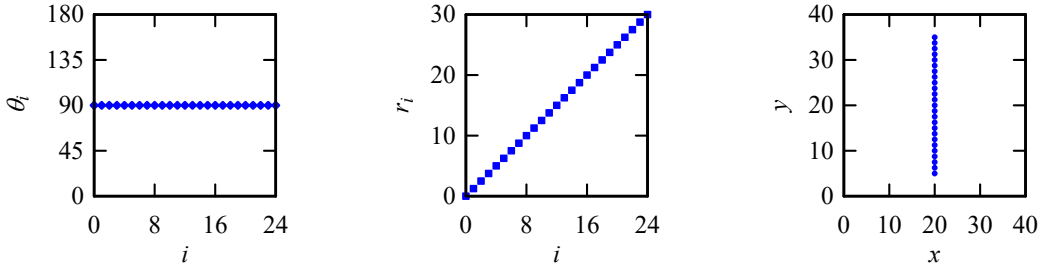
Now consider the relatively complex settings of the instantaneous distance while keeping the instantaneous rotational angle increased linearly. For example, when the instantaneous rotational angle θ_i is increased linearly, and the instantaneous distance r_i follows a tent function, the recursive process creates a point cloud *PC* representing a leaf-shape, as shown in Figure 3-10. On the other hand, when the instantaneous rotational angle θ_i increases linearly, and the instantaneous distance r_i follows a sine or cosine function with two cycles. The recursive process creates a point cloud *PC* representing an ellipse, as shown in Figure 3-11.

Similarly, when the instantaneous rotational angle θ_i increases linearly, and the instantaneous distance r_i follows a u-shape function with four cycles. The recursive process creates a point cloud *PC* representing an astroid, as shown in Figure 3-12. When the instantaneous rotational angle θ_i increases linearly, and the instantaneous distance r_i follows a bathtub function with two cycles. The recursive process creates a point cloud *PC* representing an S-shape, as shown in Figure 3-13.



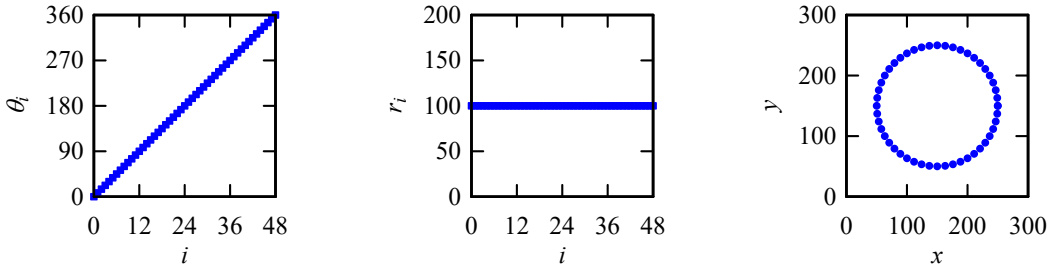
(*a*) Varible input1: θ_i (*b*) Variable input2: r_i (*c*) Output: point cloud

Figure 3-6. The setting of the recursive process to model a horizontal line.



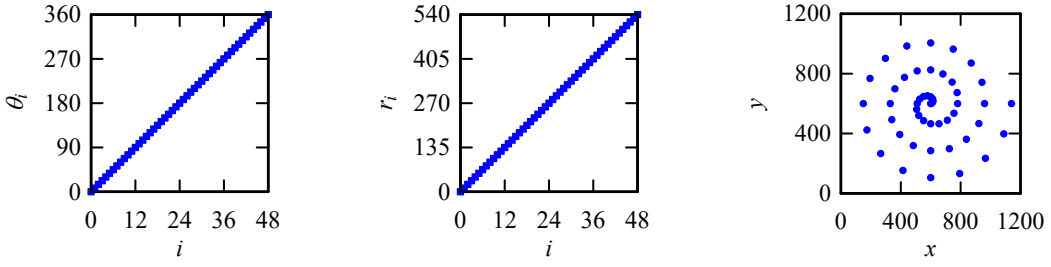
(*a*) Varible input1: θ_i (*b*) Variable input2: r_i (*c*) Output: point cloud

Figure 3-7. The setting of the recursive process to model a vertical line.



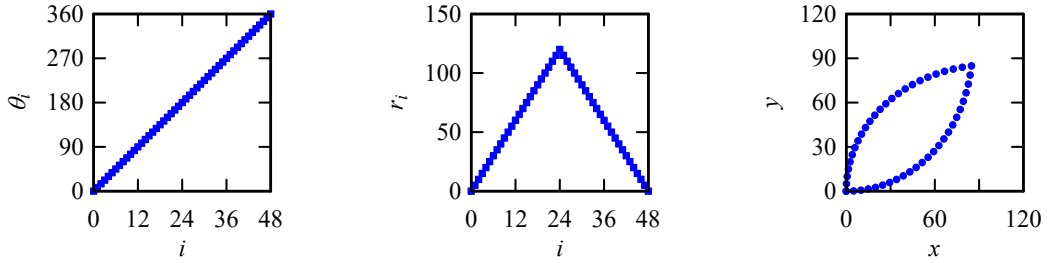
(*a*) Varible input1: θ_i (*b*) Variable input2: r_i (*c*) Output: point cloud

Figure 3-8. The setting of the recursive process to model a circle.



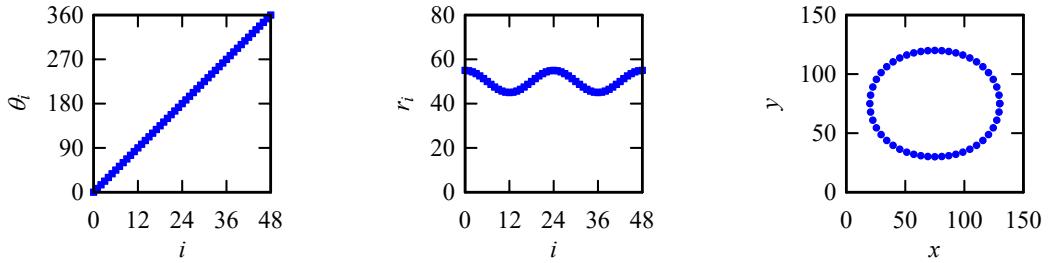
(*a*) Varible input1: θ_i (*b*) Variable input2: r_i (*c*) Output: point cloud

Figure 3-9. The setting of the recursive process to model a spiral.



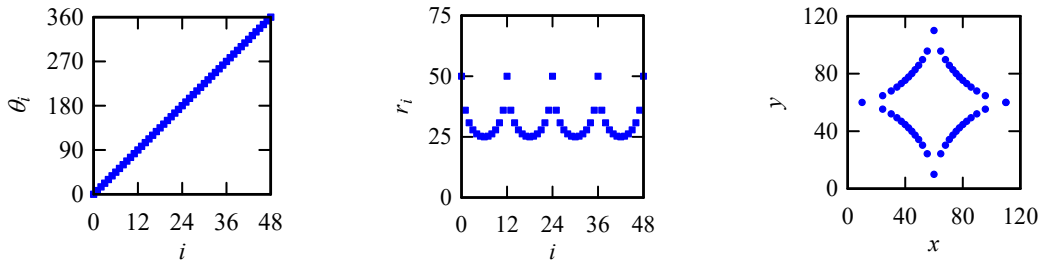
(*a*) Varible input1: θ_i (*b*) Variable input2: r_i (*c*) Output: point cloud

Figure 3-10. The setting of the recursive to model a leaf-shape.



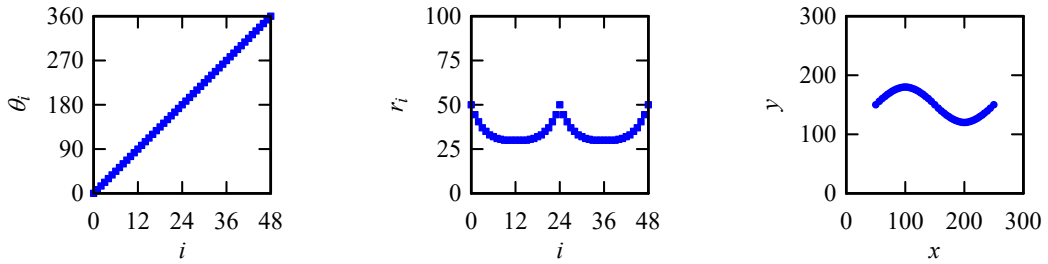
(*a*) Varible input1: θ_i (*b*) Variable input2: r_i (*c*) Output: point cloud

Figure 3-11. The setting of the recursive process to model an ellipse.



(*a*) Varible input1: θ_i (*b*) Variable input2: r_i (*c*) Output: point cloud

Figure 3-12. The setting of the recursive process to model an astroid.



(*a*) Varible input1: θ_i (*b*) Variable input2: r_i (*c*) Output: point cloud

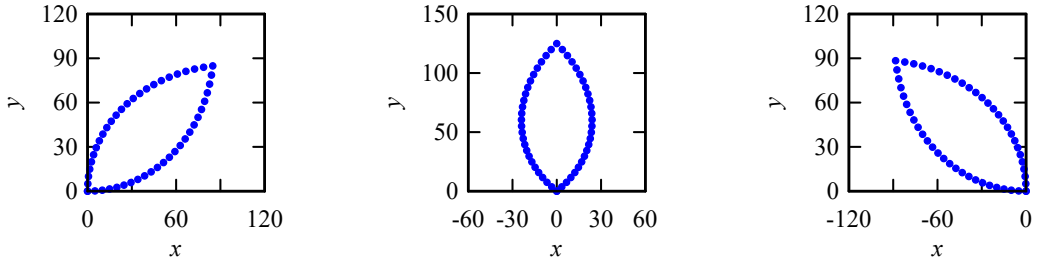
Figure 3-13. The setting of the recursive process to model an S-shape.

As shown in Figure 3-6 to Figure 3-13, keeping θ_i very simple (i.e., constant or straight-line) and varying r_i using some simple functions (e.g., straight-line, tent function, sine/cosine, u-shape function, and bathtub function) model a variety of shapes. The findings are summarized in Table 3-2. These characteristics of the recursive process can be used while modeling an arbitrary planner shape or a segment of it.

Table 3-2. Summary of the characteristics of analytical point cloud based on the variable inputs of the recursive process.

Variable Inputs		Output
Instantaneous rotational angle	Instantaneous distance	Point Cloud
Constant	Increase or decrease linearly	Straight-line
Increase linearly	Constant	Circle
Increase linearly	Increases linearly	Spiral
Increase linearly	Increase and decrease linearly like a tent function	Leaf-shape
Increase linearly	Decrease and increase gradually like a sine/cosine for two cycles	Ellipse
Increase linearly	Decrease and increase gradually like a U-shape function for four cycles	Astroid
Increase linearly	Decrease and increase gradually like a bathtub function for two cycles	S-shape

Besides the variable inputs, that is, instantaneous rotational angle and instantaneous distance, the constant input parameters, namely, the center point P_c and the initial angle ϕ, also play critical roles. The initial angle ϕ can perform the rotation function. For example, the *PC* of the leaf-shape shown in Figure 3-10 can be rotated by resetting different the initial angle ϕ values. For instance, when the initial angle ϕ is set as 0°, 45°, or 90°, the leaf-shape rotates, accordingly, as shown in Figure 3-14. Similarly, the center point can perform the translation function in both the *x*-axis and *y*-axis directions. The initial distance, on the other hand, does not play any roles. The initial distance does not affect the shape to be modeled, however, it is necessary for placing the starting point. The roles of the constant input parameters of the recursive process are summarized in Table 3-3.



(*a*) When $\phi = 0°$ (*b*) When $\phi = 45°$ (*c*) When $\phi = 90°$

Figure 3-14. Role of an initial angle.

Table 3-3. Constant input parameter roles of the recursive process.

B. Constant Inputs	Role
Center point	Translate a shape
Initial angle	Rotate shape
Initial distance	No effect on a shape

3.7. Quantification of Analytical Point Cloud

Let X be a planner shape. Let U be a method that creates a point cloud $V = \{V_i \mid i = 0, 1, \ldots\}$ to model X. Let a point cloud $PC = \{P_{ei} \mid i = 0, 1, ...\}$ is created by the recursive point process to model X. As such, both the point clouds, $V = \{V_i \mid i = 0, 1, \ldots\}$ and $PC = \{P_{ei} \mid i = 0, 1, ...\}$ must look alike. At the same time, their quantitative characteristics must be identical. Based on this consideration, this section quantifies two point clouds of the same shape using a radius of curvature and aesthetic value. For the sake of quantification, circle, ellipses, and spiral are considered, which are defined in Appendix C.

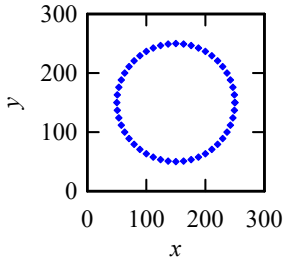
3.7.1. Quantification Based on Radius of Curvature

Let first consider the parameter called a radius of curvature, denoted as ρ_i [145]. This parameter is defined in Appendix C. A point in a point cloud has a radius of curvature, ρ_i, $i = 0, 1, \ldots$. To calculate ρ_i, the point itself and two more preceding consecutive points are needed. The variability in ρ_i is calculated for circle, ellipse, and spiral, as shown in Figure 3-15 to Figure 3-17.

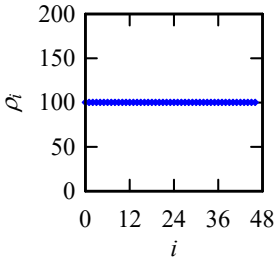
Figure 3-15 shows two point clouds of the same shape, representing a circle. Figure 3-15(*a*) shows the point cloud of the circle modeled by Equation (C.1), where the radius, $a = 100$ and center, $P_c = (150, 150)$. In this case, 49 points are created to model the circle. Figure 3-15(*b*) shows the values of ρ_i, which is equal to 100.21. Theoretically, ρ_i must be equal to $a = 100$. This small error (100.21 – 100 = 0.21) is due to calculation precision error. On the other hand, Figure 3-15(*c*) shows the point cloud of the same circle shape modeled by the recursive process. For this point cloud, $r_i = 100$ (= d); $\phi = 0$; $\theta_i = 2\pi i/n$; $n = 48$; $i = 0, 1, \ldots, 48$; and $P_c = (150, 150)$. In this case, 49 points are created to model the circle. Figure 3-15(*d*) shows the values of ρ_i, which is equal to 100.21. It means that r_i plays the role of the radius of curvature of a circle.

Figure 3-16 shows two point clouds of the same shape, representing an ellipse. Figure 3-16(*a*) shows the point cloud of the ellipse modeled by Equation (C.2), where the major radius, $a = 55$; minor radius, $b = 45$; and center, $P_c = (75, 75)$. In this case, 49 points are created to model the ellipse. Figure 3-16(*b*) shows the variation of ρ_i. On the other hand, Figure 3-16(*c*) shows the point cloud of the same ellipse shape modeled by the recursive process. For this point cloud, $d = 15$; $\phi = 0$; r_i is decreased linearly from 55 to 45 and increased linearly from 45 to 55 for two cycles; $\theta_i = 2\pi i/n$; $n = 48$; $i = 0, 1, \ldots, 48$; and $P_c = (75, 75)$. In this case, as well, 49 points are created to model the ellipse. Figure 3-16(*d*) shows the variation of ρ_i. As seen in Figure 3-16 and Figure 3-16, both point clouds are the same, and their radius of curvature also varies in the same trend. It means that r_i plays the role of the radius of curvature of an ellipse.

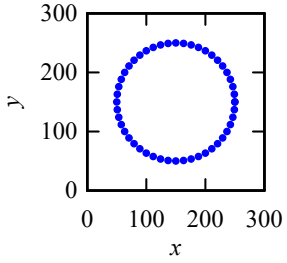
Figure 3-17 shows two point clouds of the same shape, representing a spiral. Figure 3-17(*a*) shows the point cloud of the spiral shape modeled by Equation (C.3), where an arbitrary, positive, real constant, $a = 28.647$ that controls the distance between the successive turnings of the spiral. In this case, 49 points are created to model the spiral. Figure 3-17(*b*) shows the variation of ρ_i. On the other hand, Figure 3-17(*c*) shows the point cloud of the same spiral shape modeled by the recursive process. For this point cloud, $d = 15$; $\phi = 0$; $r_i = 0, 11.25, \ldots, 540$; $\theta_i = 2\pi i/n$; $n = 48$; $i = 0,1, \ldots, 48$; and $P_c = (600, 600)$. In this case, as well, 49 points are created to model the spiral. Figure 3-17(*d*) shows the variation of ρ_i. As seen in Figure 3-17 and Figure 3-17, both point clouds are the same, and their radius of curvature also varies in the same trend. It means that r_i plays the role of a radius of curvature of a spiral.



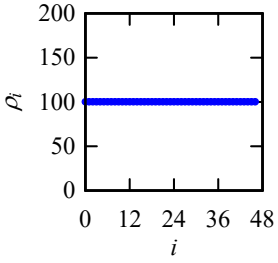
(*a*) Point cloud created by Equation (C.1)



(*b*) The radius of curvature of (*a*)

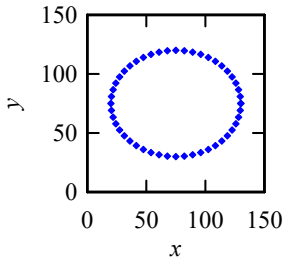


(*c*) Point cloud created by the recursive process

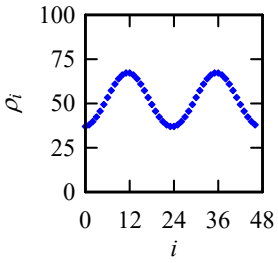


(*d*) The radius of curvature of (*c*)

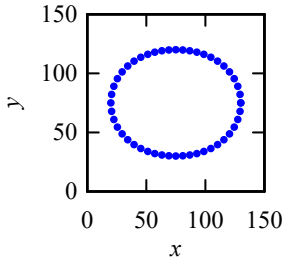
Figure 3-15. Circle shapes and their radii of curvature.



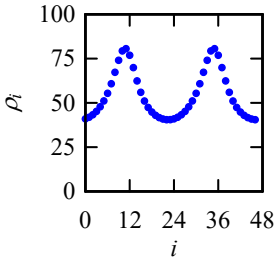
(*a*) Point cloud created by Equation (C.2)



(*b*) The radius of curvature of (*a*)

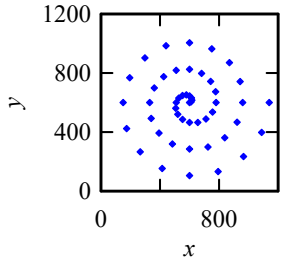


(*c*) Point cloud created by the recursive process

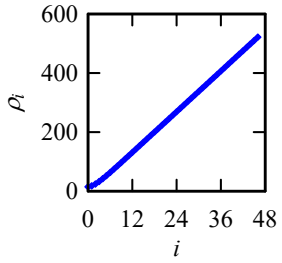


(*d*) The radius of curvature of (*c*).

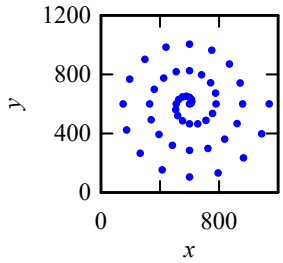
Figure 3-16. Ellipse shapes and their radii of curvature.



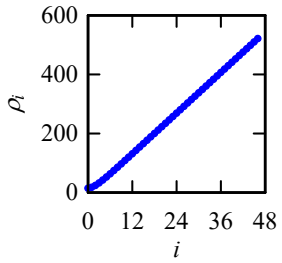
(*a*) Point cloud created by Equation (C.3)



(*b*) The radius of curvature of (*a*)



(*c*) Point cloud created by the recursive process



(*d*) The radius of curvature of (*c*)

Figure 3-17. Spiral shapes and their radii of curvature.

It is clear from Figure 3-15 to Figure 3-17, the shape modeled by the parametric equation-based and the recursive process is visually identical. It shows that the instantaneous distance plays the role of the radius of curvature when the shapes are modeled by the recursive process. The instantaneous distance variation is comparatively lower than the radius of curvature that shows the geometry of the shape can be easily controlled using the instantaneous distance compared to the radius of curvature. It is worth mentioning that there are several critical elements for shape designing. The mathematical formulation of a radius of curvature or instantaneous distance can be one effective approach [146].

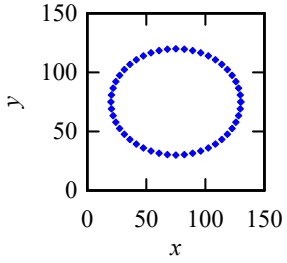
3.7.2. Quantification Based on the Aesthetic Value

This subsection describes the quantification of shapes based on an aesthetic value parameter, denoted as α [147-149]. This parameter is defined in Appendix C. To understand the aesthetic value, a plot called a logarithm curvature histogram (LCH) is used. The procedure to construct LCH for a given curve (in this case, from a given point cloud) is also shown in Appendix C. For quantification of the aesthetic value, the ellipse and spiral shape are considered.

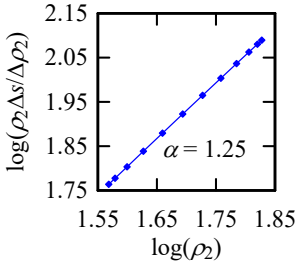
Figure 3-18(*a*) shows the point cloud of the ellipse modeled by Equation (C.2), which is repeated from the previous section (Figure 3-16(*a*)). The corresponding LCH of the ellipse is shown in Figure 3-18(*b*). Figure 3-18(*c*) shows the point cloud of the ellipse modeled by the recursive process (taken from Figure 3-16(*c*)). The corresponding LCH of the ellipse is shown in Figure 3-18(*d*). Both point clouds have the same aesthetic value $\alpha = 1.25$.

Figure 3-19(*a*) shows the point cloud of a spiral modeled by Equation (C.3) (taken from Figure 3-17(*a*)). The corresponding LCH of the spiral is shown in Figure 3-19(*b*). Figure 3-19(*c*) shows the point cloud of the spiral modeled by the recursive process (repeated from Figure 3-17(*c*)). The corresponding LCH of the spiral is shown in Figure 3-19(*d*). Both point clouds of the spiral shape have the same aesthetic value $\alpha = 1.22$.

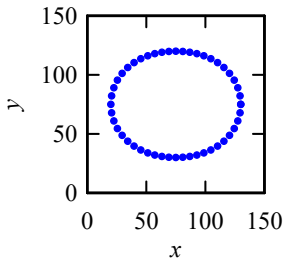
As seen in Figure 3-18 and Figure 3-19, even though the shapes are created using the recursive process instead of the respective parametric equation-based, their aesthetic natures remain unaffected. It is also observed that the length of the line segment in the LCH depends on the variation range (minimum value to maximum value) of the radius of curvature. If the variation in the radius of curvature is high, the length of LCH becomes larger and vice versa.



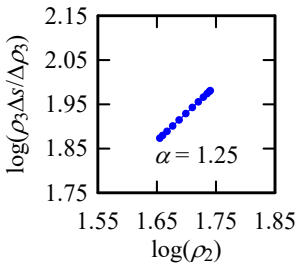
(*a*) Point cloud created by Equation (C.2)



(*b*) Logarithm curvature histogram of (*a*)

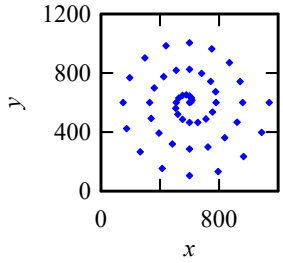


(*c*)Point cloud created by the recursive process

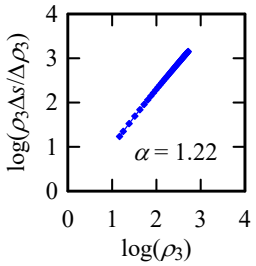


(*d*) Logarithm curvature histogram of (*c*)

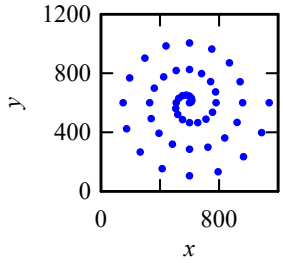
Figure 3-18. Logarithm curvature histogram of ellipses.



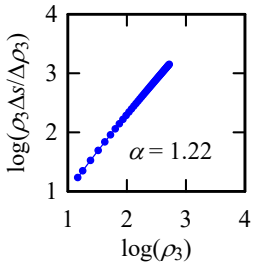
(*a*) Point cloud created by a parametric equation (C.3)



(*b*) Logarithm curvature histogram of (*a*)



(*c*) Point cloud created by the recursive process



(*d*) Logarithm curvature histogram of (*c*)

Figure 3-19. Logarithm curvature histogram of spirals.

This chapter addresses the methodology of this study. The chapter presented the framework of the proposed reverse engineering and described each process of the proposed reverse engineering method in detail. The chapter also presented the characteristics and quantified the analytical point cloud using the radius of curvature and aesthetic value. The next chapter will address the applications of the proposed reverse engineering method.

Chapter 4: Applications

This chapter addresses the applications of the proposed reverse engineering method. Applications demonstrate the modeling efficacy of the proposed reverse engineering method. Section 4.1 presents the digitization of basic Hokkaido Ainu motifs and their physical prototyping. Section 4.2 presents the digitization of complex Hokkaido Ainu patterns and their physical prototyping. Section 4.3 presents the digitization and physical prototyping of an ancient ewer.

4.1. Digitization and Physical Prototyping of Ainu Motifs

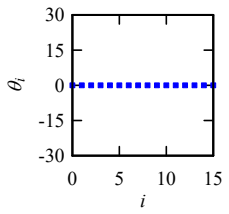
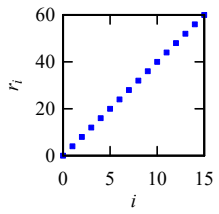
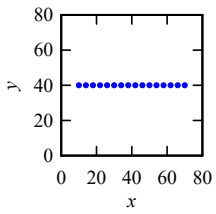
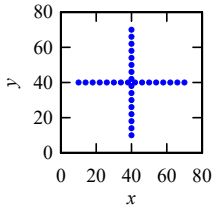
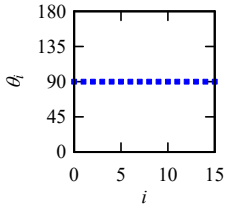
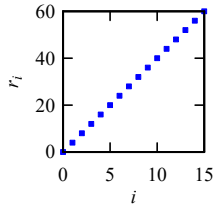
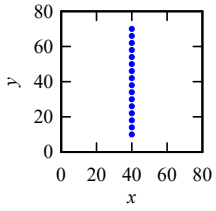
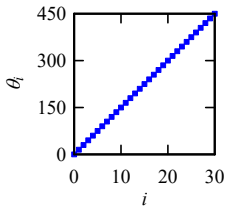
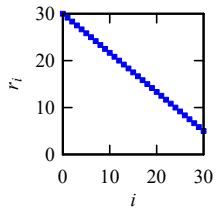
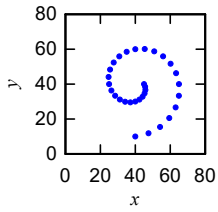
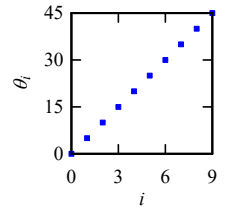
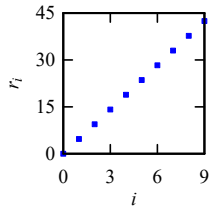
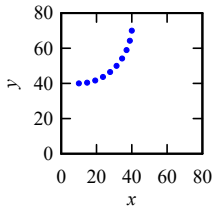
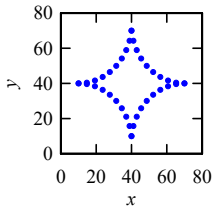
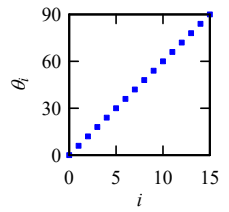
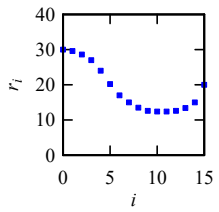
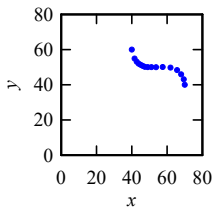
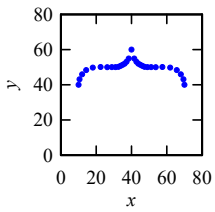
This section presents the digitizing results of the main, synthetic, and plant motifs of Hokkaido Ainu. First, the point clouds of some selected main, synthetic, and plant motifs are presented. Afterward, the virtual and physical prototypes of the main, synthetic, and plant motifs are presented.

4.1.1. Point Clouds of the Main Motifs

There are four main motifs, namely, Utasa, Morew, Sik, and Ayus, as shown in Table 2-3. Table 4-1 shows the point cloud of four main motifs and the settings of the instantaneous rotational angles and instantaneous distances.

As seen in Table 4-1, to model Utasa, two point clouds were needed. One of the point clouds represents a vertical line, and the other point cloud represents a horizontal line. The point cloud of the horizontal line is created by keeping the instantaneous rotational angles constant at 0°, and the instantaneous distances were increased linearly. The point cloud of the vertical line was created by keeping the instantaneous rotational angles constant at 90°, and the instantaneous distances were increased linearly. These two point clouds were integrated to create the point cloud of Utasa, as shown in the last column in Table 4-1.

Table 4-1. Point clouds and the settings of the instantaneous distances and instantaneous rotational angles of the main motifs of Hokkaido Ainu.

Motif Name	Instantaneous Rotational Angle	Instantaneous Distance	Elementary Point Cloud	Coordinated Point Cloud
Utasa				
				
Morew				---
Sik				
Ayus				

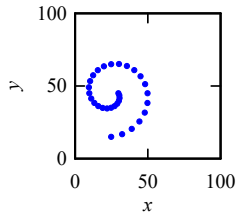
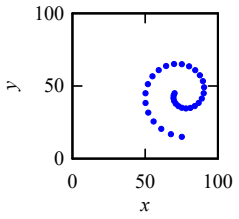
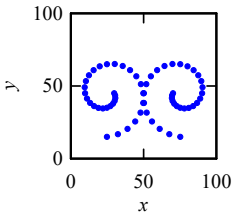
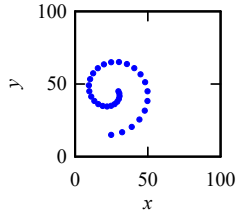
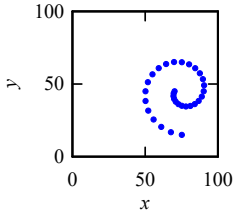
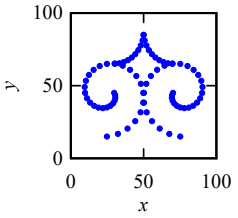
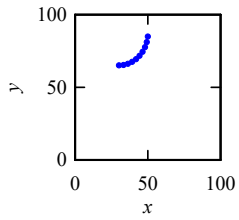
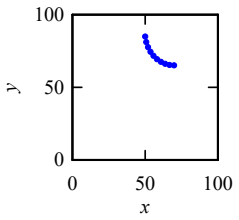
For Morew, the point cloud was created by linearly increasing the instantaneous rotational angles and linearly decreasing the instantaneous distances. A single point cloud is enough to model Morew. To model Sik, four point clouds were needed. The first point cloud was created by linearly increasing both the instantaneous rotational angles and the instantaneous distances. Other three point clouds were created by copying and rotating the first point cloud, as shown in the last column in Table 4-1. To model Ayus, two point

clouds were needed. The first point cloud was created by linearly increasing the instantaneous rotational angles. The instantaneous distances were first decreased and then increased nonlinearly in a systematic manner. The other point cloud was created by inverting the first point cloud, as shown in the last column in Table 4-1.

4.1.2. Point Clouds of Synthetic Motifs

There are four first-order synthetic motifs, namely, Uren-morew, Ski-uren-morew, Arus-morew, and Sikikew-nu-morew, as shown in Table 2-3. Out of these four motifs, two motifs, Uren-morew and Ski-uren-morew, have been selected. The point clouds of the synthetic motifs can be created using the point cloud of the main motifs. It means that the point clouds shown in Table 4-1 can create the point clouds of two selected synthetic motifs, as shown in Table 4-2.

Table 4-2. Point clouds of two selected synthetic motifs of Hokkaido Ainu.

Motif Name	Elementary Point Cloud	Inverted Point Cloud	Coordinated Point Cloud
Uren-morew			
Ski-uren-morew			
			

As seen in Table 4-2, to model the synthetic motif called Uren-morew, two point clouds were used. One of the point clouds was the point cloud of Morew, as shown in

Table 4-1. The other point cloud was created by inverted the point cloud of Morew. On the other hand, to model the synthetic motif called Ski-uren-morew, four point clouds were used. The first point cloud was a quarter-point cloud of Sik, and the second point cloud was the point cloud of Morew. Integrating the first and second point clouds results in the point cloud of half Uren-morew. The other half was created by inverting the point cloud of the first half, as shown in Table 4-2.

4.1.3. Point Clouds of the Plant Motifs

There are four plant motifs, namely, Morew-etok, Punkar, Apapo-piras (u) ke, and Apapo-epuy, as shown in Table 2-3. Morew-etok, Apapo-piras (u) ke, and Apapo-epuy are selected as shown in Table 4-3. The point cloud of the Punkar motif can be created using a similar procedure.

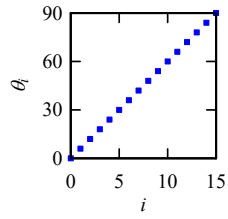
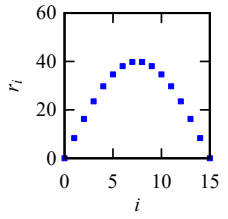
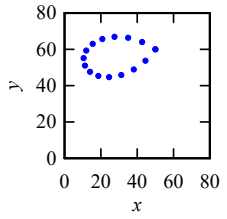
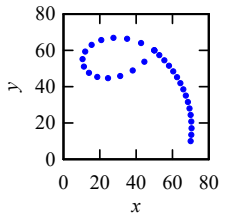
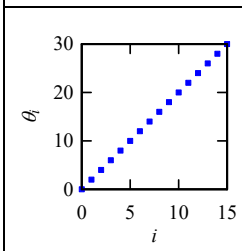
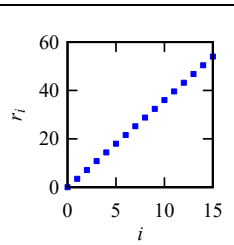
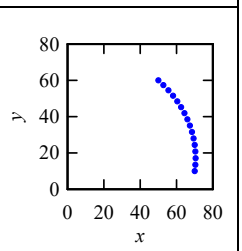
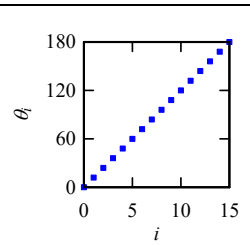
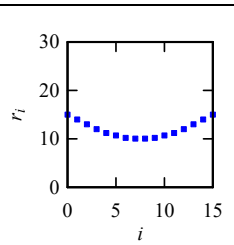
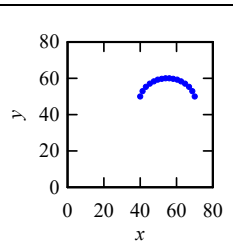
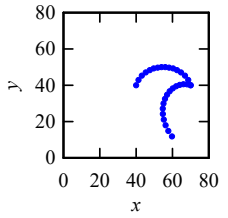
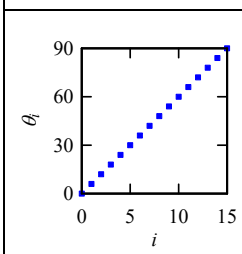
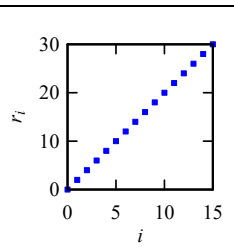
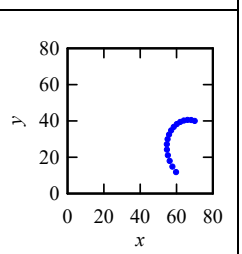
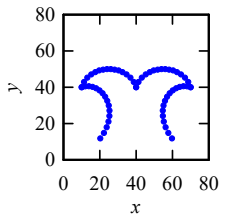
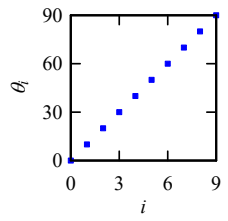
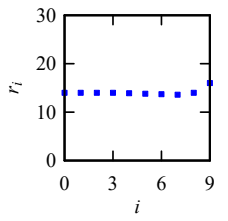
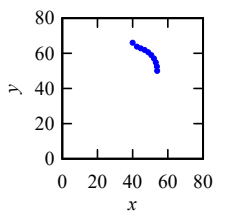
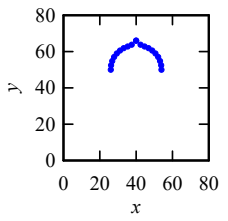
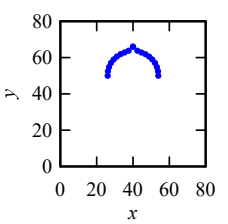
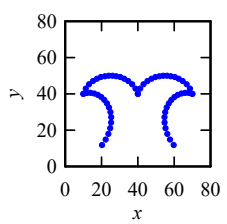
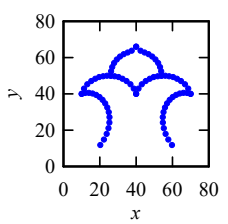
As shown in Table 4-3, the plant motif called Morew-etok was modeled using two point clouds. The first point cloud is created by linearly increasing the instantaneous rotational angles, and the instantaneous distances were varied in dome-like function. The recursive process created a point cloud representing a leaf-like shape. The other point cloud (that is a tail) was created by increasing the instantaneous rotational angles and distances linearly, as shown in the first row in Table 4-3.

The plant motif called Apapo-piras (u) ke was modeled using four point clouds, as shown in Table 4-3. The first point cloud was created by linearly increasing the instantaneous rotational angles, and the instantaneous distances were varied using a concave function with the point inflection in the middle of the iterations. The second point cloud was created by linearly increasing both the instantaneous rotational angles and the instantaneous distances. Integrating the first and second point clouds results in the point cloud of half Apapo-piras (u) ke. The other half was created by inverting the point cloud of the first half, as shown in Table 4-3.

The plant motif called Apapo-epuy was modeled by placing an elliptical-shape motif on the top of the Apapo-piras (u) ke, as shown in the last row in Table 4-3. First, the point cloud of the half segment of the elliptical-shape motif was created by linearly increasing the instantaneous rotational angles, and keeping the instantaneous distances almost constant for all the iterations. The other half segment was created by inverting the

first half segment. Then the point cloud of the elliptical-shape motif and the point cloud of Apapo-piras (u) ke were integrated, as shown in the last column in Table 4-3.

Table 4-3. Point cloud and the setting of the instantaneous distances and instantaneous rotational angles of two selected plant motifs of Hokkaido Ainu.

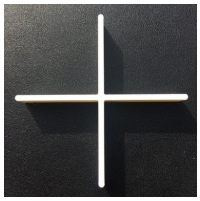
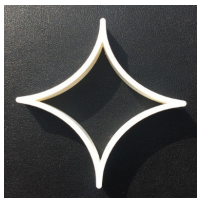
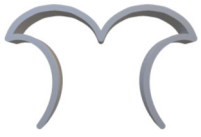
Motif Name	Instantaneous Rotational Angle	Instantaneous Distance	Elementary Point Cloud	Coordinated Point Cloud
Morew-etok				
				
Apapo-piras (u) ke				
				
Apapo-epuy				
				

It is worth mentioning that when creating the point cloud, it is better to begin from the center of a model and go outward. This step helps calculate the size of the shape and its location on the x-y plane.

4.1.1. Virtual and Physical Models of Ainu Motifs

Table 4-4 shows ten selected virtual models and physical objects of basic Hokkaido Ainu motifs. All the virtual models were created by inputting the respective point clouds to a commercially available 3D CAD system, as prescribed by the methodology presented in Chapter 3. The triangulation data of the virtual models were used for building the physical models. The physical objects were built by using a 3D printer available at Kitami Institute of Technology.

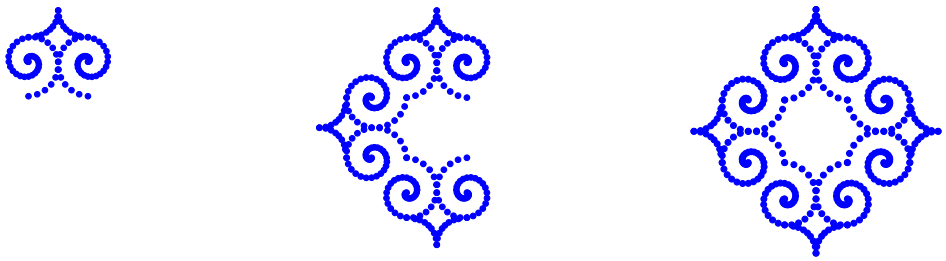
Table 4-4 Solid CAD models and 3D printed objects of the basic Hokkaido Ainu motifs.

Motif Names	Virtual Models	Physical Object	Motif Name	Virtual Models	Physical Object
Utasa			Ski-uren-morew		
Morew			Morew-etok		
Sik			Punkar		
Ayus			Apapo-piras (u) ke		
Uren-morew			Apapo-epuy		

4.2. Digitization and Physical Prototyping of Ainu Patterns

When Hokkaido Ainu patterns crafted on houses, clothing, ornaments, utensils, and spiritual goods are studied, complex patterns are observed. The complex patterns can be created by synthesizing one or more basic motifs. In this study, six selected complex patterns denoted as Pattern-1, Pattern-2, Pattern-3, Pattern-4, Pattern-5, and Pattern-6 were digitized using the methodology presented in Chapter 3. The descriptions are as follows.

Pattern-1 was created based on the synthetic motif called Ski-uren-morew, as shown in Figure 4-1. Pattern-1 consists of four pieces of the Ski-uren-morew motifs. The settings of the instantaneous rotational angle and instantaneous distance to the created point cloud have been described in Section 4.1. The point cloud of Ski-uren-morew motifs was rotated three times using Equation (C.4), where $\theta = \pi/2$, π, and $3\pi/2$. Then, all the point clouds were integrated into a single point cloud, which results in Pattern-1. The point cloud was then transferred to a 3D CAD system to create a virtual model (a solid CAD model). The triangulation data of the virtual model were used to produce the physical object using a commercially available 3D printer, as shown in Figure 4-2.

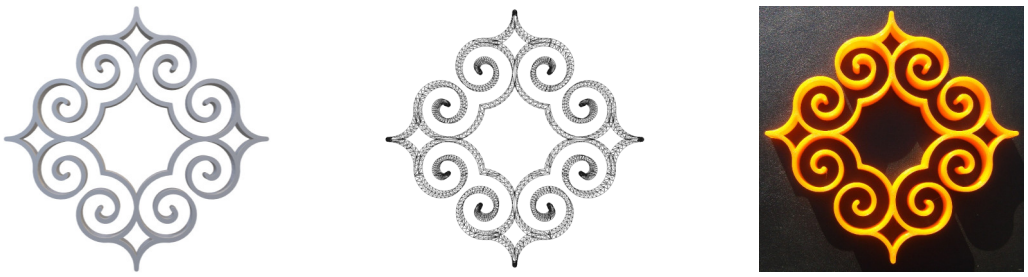


(*a*) Point cloud of the Ski-uren-morew motif

(*b*) Point created after rotating two times

(*c*) Point created after rotating three times

Figure 4-1. Point cloud coordination of Pattern-1 based Hokkaido Ainu motif.



(*a*) Virtual model

(*b*) Triangulated model

(*c*) 3D printed object

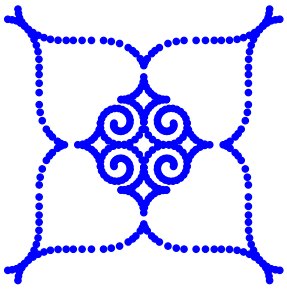
Figure 4-2. Virtual model, triangulated model, and 3D printed object of Pattern-1.

Pattern-2 was created based on three motifs called Ski-uren-morew, Sik, and Ayus, as shown in Figure 4-3. The settings of the instantaneous rotational angle and instantaneous distance to the created point cloud have been described in Section 4.1. Pattern-2 has an internal segment and an external segment. For the internal segment, the point clouds of two Ski-uren-morew motifs were placed horizontally. The point clouds of two Sik motifs were placed vertically. The external segment consists of four symmetrically integrated point cloud of Ayus motifs, which results in Pattern-2. The result is shown in Figure 4-3(*c*). The virtual model, triangulated model, and physical prototype were created similarly to Pattern-1. The results are shown in Figure 4-4.

(*a*) Point cloud of the Ski-uren-morew motif

(*b*) Point cloud of two Ski-uren-morew motifs placed horizontally

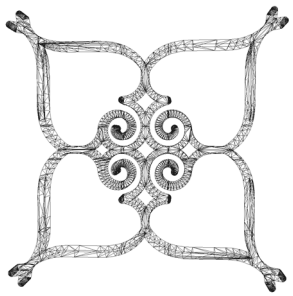
(*c*) Point cloud of two Siks motifs placed vertically



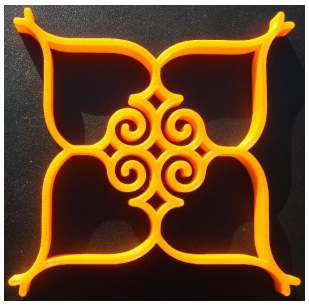
(*d*) Point cloud of four Ayus motifs placed externally

Figure 4-3. Point cloud coordination of Pattern-2 based Hokkaido Ainu motif.

(*a*) Virtual model



(*b*) Triangulated model

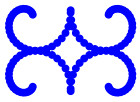


(*c*) 3D printed object

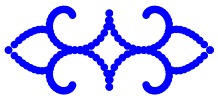
Figure 4-4. Virtual model, triangulated model, and 3D printed object of Pattern-2.

Pattern-3 was created based on three motifs called Uren-morew, Sik, and elliptically-shaped motifs, as shown in Figure 4-5. The settings of the instantaneous rotational angle and instantaneous distance to the created point cloud have been described in Section 4.1. The point clouds of two Uren-morew motifs were placed symmetrically. The point clouds of two elliptically-shaped motifs were placed adjacent to Uren-morew motifs. Also, the point clouds of two Sik motifs were placed vertically in a symmetrical manner, which results in Pattern-3. The result is shown in Figure 4-5(*d*). The virtual model, triangulated model, and physical prototype were created similarly to Pattern-1. The results are shown in Figure 4-6.

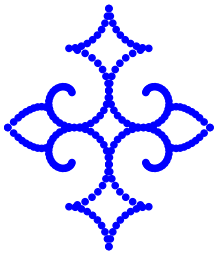
(*a*) Point cloud of the Uren-morew motif



(*b*) Point cloud of two Uren-morew motifs



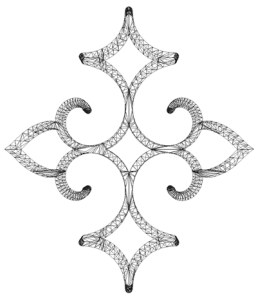
(*c*) Point cloud of two elliptical-shaped motifs placed adjacent to the Uren-morew motifs



(*d*) Point cloud of two Sik motifs placed vertically

Figure 4-5. Point cloud coordination of Pattern-3 based Hokkaido Ainu motif.

(*a*) Virtual model



(*b*) Triangulated model

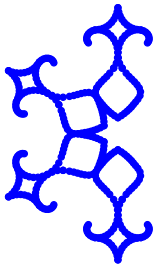
(*c*) 3D printed object

Figure 4-6. Virtual model, triangulated model, and 3D printed object of Pattern-3.

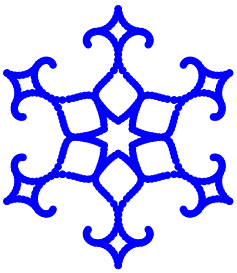
Pattern-4 was created based on two motifs called Ski-uren-morew and Sik, as shown in Figure 4-7. The settings of the instantaneous rotational angle and instantaneous distance to the created point cloud have been described in Section 4.1. The point cloud of the Ski-uren-morew motif was placed adjacent to the point cloud of the elliptically shaped motif. The point clouds were then copied and rotated five times using Equation (C.4), defined in Appendix C, which results in Pattern-4. The result is shown in Figure 4-7(*d*). The virtual model, triangulated model, and physical prototype was created similarly to Pattern-1. The results are as shown in Figure 4-8.

(*a*) Point cloud of the Ski-uren-morew motif

(*b*) Point cloud of elliptically-shaped motif placed adjacent to the point cloud of the Ski-uren-morew motif



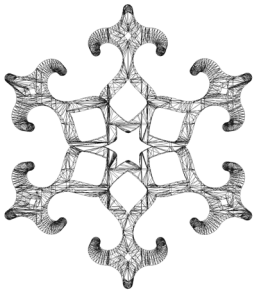
(*c*) Point cloud created after rotating three times



(*d*) Point cloud created after rotating five times

Figure 4-7. Point cloud coordination of Pattern-4 based Hokkaido Ainu motif.

(*a*) Virtual model

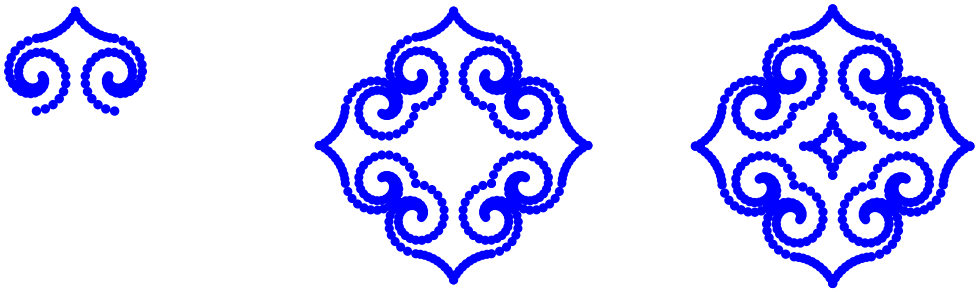


(*b*) Triangulated model

(*c*) 3D printed object

Figure 4-8. Virtual model, triangulated model, and 3D printed object of Pattern-4.

Pattern-5 was created based on two motifs called Ski-uren-morews and Sik, as shown in Figure 4-9. The settings of the instantaneous rotational angle and instantaneous distance to the created point cloud of the Ski-uren-morew motif have been described in Section 4.1. Pattern-5 consists of four Ski-uren-morew motifs and a Sik motif. First, the point cloud of the Ski-uren-morew motif was created. The point cloud of the other three Ski-uren-morew motifs was created by rotating the first Ski-uren-morew. The rotation was performed using Equation (C.4), where $\theta = \pi/2$, π, and $3\pi/2$. Then, the point cloud of the Sik motif was created. Sik motif's point cloud was placed inside the point clouds of four Ski-uren-morew motifs, which results in Pattern-5. The result is shown in 4-9(*c*). The virtual model, triangulated model, and physical prototype was created similarly to Pattern-1. The results are as shown in Figure 4-10.

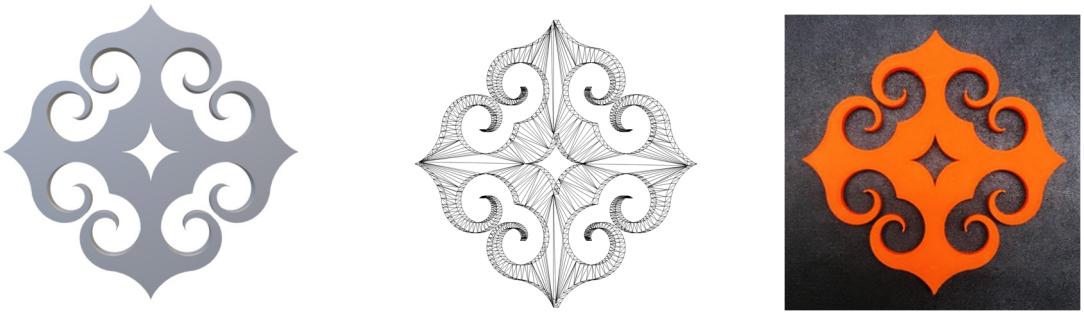


(*a*) Point cloud of the Ski-uren-morew

(*b*) Point cloud created after rotating four times

(*c*) Point cloud of the Sik motifs placed inside the Ski-uren-morews motifs

Figure 4-9. Point cloud coordination of Pattern-5 based Hokkaido Ainu motif.



(*a*) Virtual model

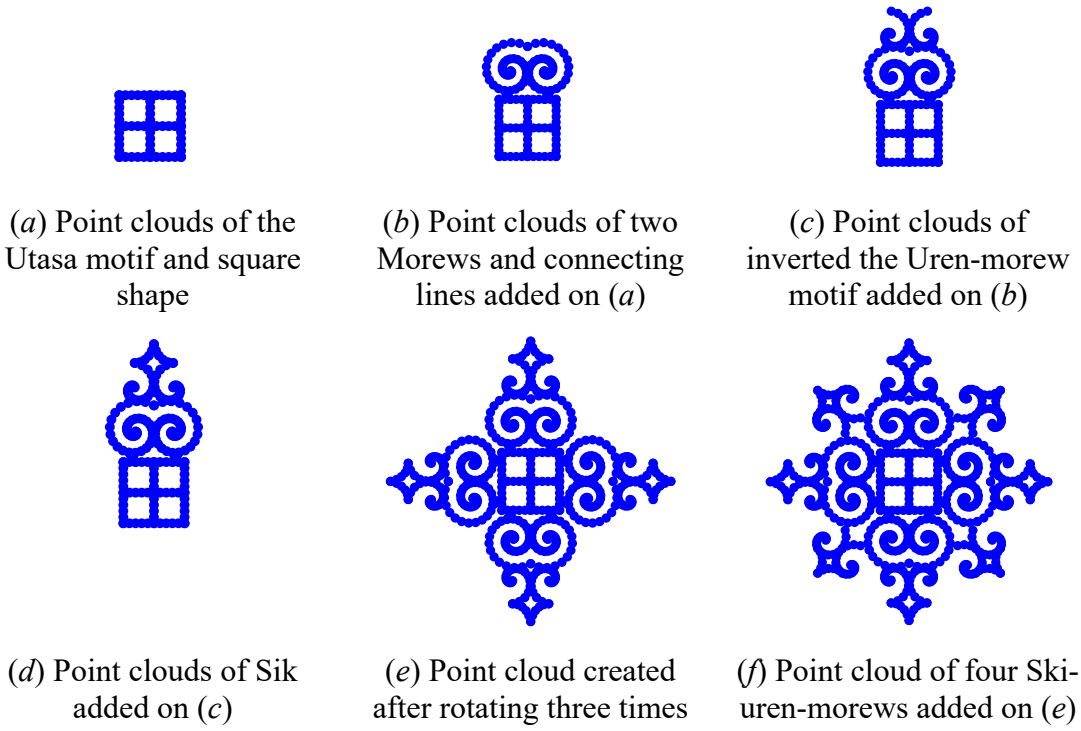
(*b*) Triangulated model

(*c*) 3D printed object

Figure 4-10. Virtual model, triangulated model, and 3D printed object of Pattern-5.

Pattern-6 was created based on five motifs called Morew, Sik, Utasa, inverted Uren-morew, and Ski-uren-morew, as shown in Figure 4-11. First, the point clouds of Utasa and square shape motifs were created. Then, the point clouds of the two Morew motifs were created to become symmetric about the *y*-axis. Next, the point clouds of two lines

were created to connect the point clouds of Morew motifs to the point cloud of a square shape. The same was done for the two point other point clouds of inverted Uren-morew motifs, which are symmetrical about the *y*-axis. Then, the point cloud of the Sik motifs was created on the top of the inverted Uren-morew motif's point cloud. Now point clouds of two connecting lines, Morews, inverted Uren-morew, and Sik motifs were rotated using Equation (C.4), where $\theta = \pi/2$, π, and $3\pi/2$. Next, the point cloud of four Ski-uren-morew motifs was created and then placed between the Morew motifs, which results in Pattern-6. The result is shown in Figure 4-11(*f*). The virtual model, triangulated model, and physical prototype was created similarly to Pattern-1, as shown in Figure 4-12.



(*a*) Point clouds of the Utasa motif and square shape

(*b*) Point clouds of two Morews and connecting lines added on (*a*)

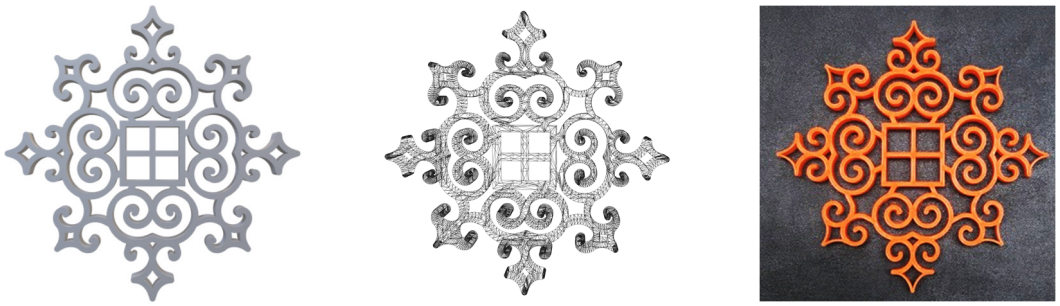
(*c*) Point clouds of inverted the Uren-morew motif added on (*b*)

(*d*) Point clouds of Sik added on (*c*)

(*e*) Point cloud created after rotating three times

(*f*) Point cloud of four Ski-uren-morews added on (*e*)

Figure 4-11. Point cloud coordination of Pattern-6 based Hokkaido Ainu motif.



(*a*) Virtual model

(*b*) Triangulated model

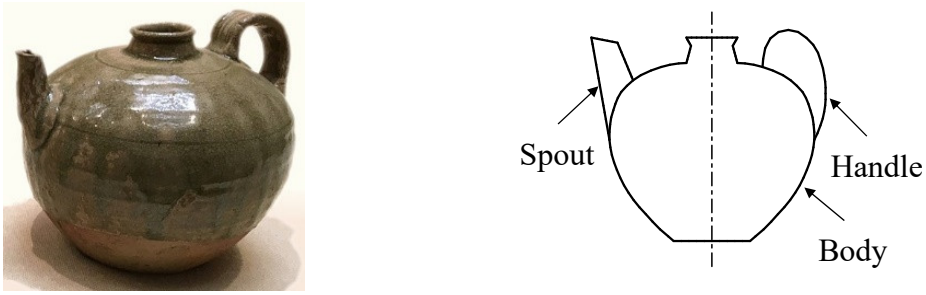
(*c*) 3D printed object

Figure 4-12. Virtual model, triangulated model, and 3D printed object of Pattern-6.

4.3. Digitization and Physical Prototyping of an Ancient Ewer

In addition to the Hokkaido Ainu motifs, the proposed reverse engineering method can be implemented in the digitization and physical prototyping of other significant cultural artifacts/objects. It was demonstrated by prototyping a virtual model and physical object (replica) of an ancient ewer, shown in Figure 4-13. The ancient ewer has a 3D shape compared to the Hokkaido Ainu motifs, which has a 2.5D shape. This demonstrates that the proposed reverse engineering method can be used for modeling both for a planner shape (2.4D shape) and a complex shape (3D shape).

The ancient ewer is shown in Figure 4-13(*a*) is from Seto ware (Seto is a city located in Aichi Prefecture, Japan) made during the Nanboku-chō period in the 13th century [150]. The ewer consists of three parts: body, spout, and handle, as shown in Figure 4-13(*b*). As seen in Figure 4-13(*b*), a few line segments and a concave can represent the body, line segments can represent the spout, and an elliptical shape can represent the handle. Hence, the ewer can be constructed using line segments, concave curves, and ellipse.



(*a*) Photograph of an ewer [150] (b) Illustration of (a)

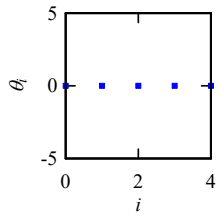
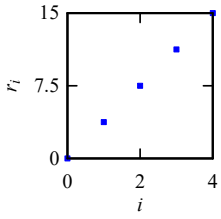
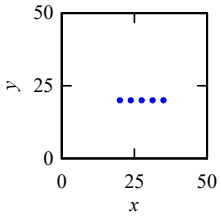
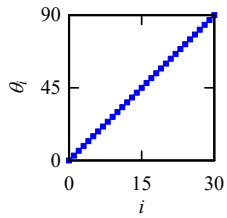
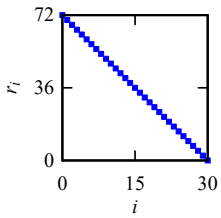
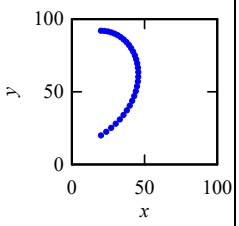
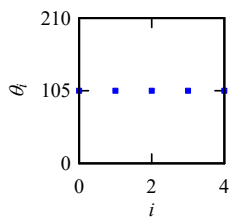
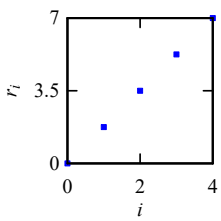
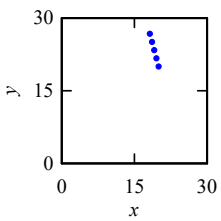
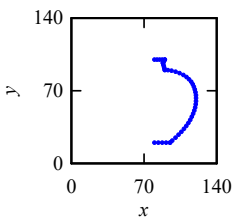
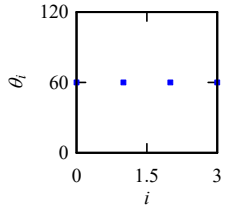
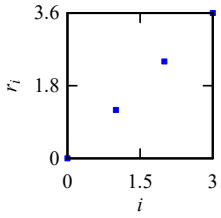
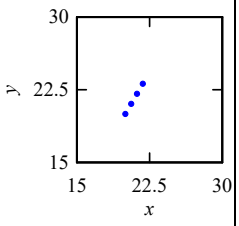
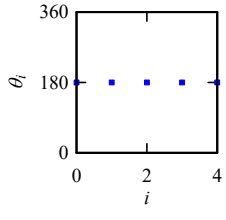
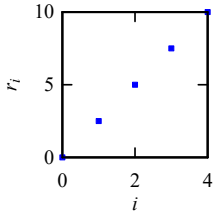
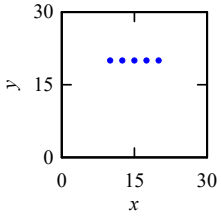
Figure 4-13. An ancient ewer.

4.3.1. Point cloud Creation of Ewer Body

Table 4-5 shows the elementary point clouds and a coordinated point cloud of the ewer half body. As seen in Table 4-5, to model the ewer half body, point clouds of four line segments and a concave curve were needed. The point cloud of the line segments was created by keeping the instantaneous rotational angles constant at certain angles. The instantaneous distances were increased linearly. The instantaneous rotational angles determine the slope of the line segments. The concave curve's point cloud was created by linearly increasing the instantaneous rotational angles and linearly decreasing the

instantaneous distances. Afterward, all the segments' elementary point clouds were joined sequentially based on the order of features, as shown in the last column in Table 4-5.

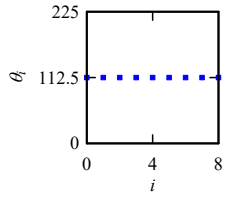
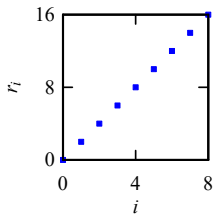
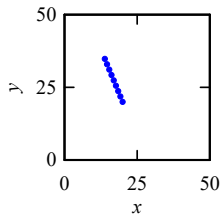
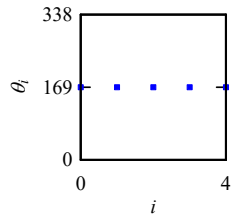
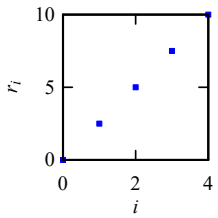
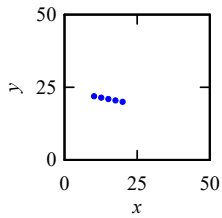
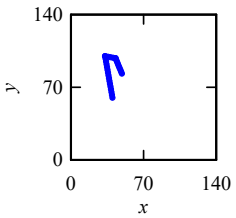
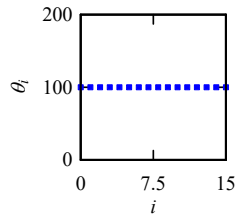
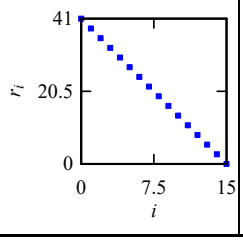
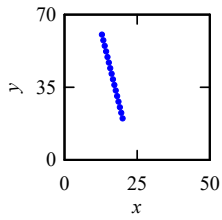
Table 4-5. The settings of the recursive process to model the ewer half body.

Segment	Instantaneous Rotational Angle	Instantaneous Distance	Elementary Point Cloud	Coordinated Point Cloud
1				
2				
3				
4				
5				

4.3.2. Point Cloud Creation of Ewer Spout

Table 4-6 shows the elementary point clouds and a coordinated point cloud of the ewer spout. As seen in Table 4-6, to model the ewer spout, point clouds of three line segments were needed. The point cloud of the line segments was created by keeping the instantaneous rotational angles constant at certain angles. The instantaneous distances were increased linearly. The instantaneous rotational angles determine the slope of the line segments. Afterward, all the line segments' elementary point clouds were joined sequentially based on the order of features, as shown in the last column in Table 4-6.

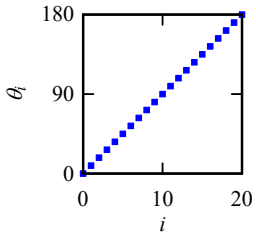
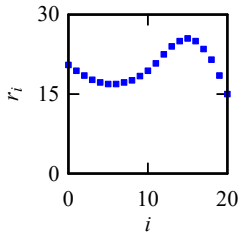
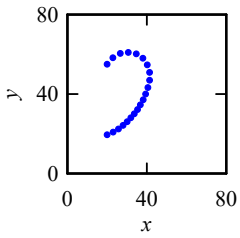
Table 4-6. The settings of the recursive process to model the ewer spout.

Segment	Instantaneous Rotational Angle	Instantaneous Distance	Elementary Point Cloud	Coordinated Point Cloud
1				
2				
3				

4.3.3. Point Cloud Creation of Ewer Handle

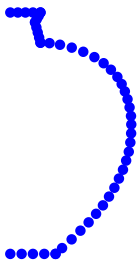
Table 4-7 shows the elementary point clouds the ewer handle. As seen in Table 4-7, the point cloud of the elliptical shape was created by linearly increasing the instantaneous rotational angles, and the instantaneous distances were varied in a cosine function. In the ewer handle, a single elementary point cloud is enough to model the ewer's handle.

Table 4-7. The settings of the recursive process to model the ewer handle.

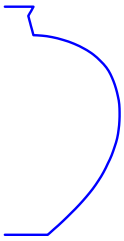
Segment	Instantaneous Rotational Angle	Instantaneous Distance	Elementary Point Cloud
1			

4.3.4. Solid CAD Modeling of Ewer Body

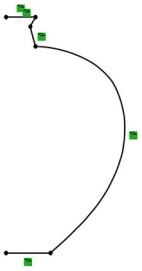
Figure 4-14 shows the results of solid CAD modeling of the ewer body. As shown in Figure 4-14, the ewer half body's point cloud was exported to a 3D CAD system, applied entity, constructed centerline, and then revolved around the centerline. The result of solid CAD modeling is shown in Figure 4-14(*c*). The revolve was performed using the revolve function available in the 3D CAD system. In this study, Solid CAD modeling was performed using SolidWorks.



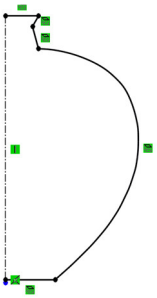
(*a*) Point cloud of the ewer half body



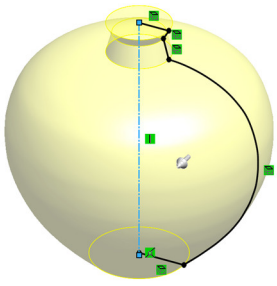
(*b*) Point cloud after exporting to a 3D CAD system



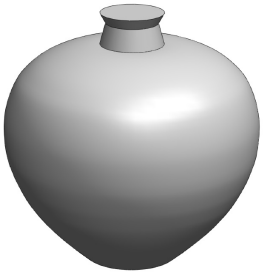
(*c*) Entity conversion



(*d*) Draw the centerline to revolve



(*e*) Revolve around the centerline



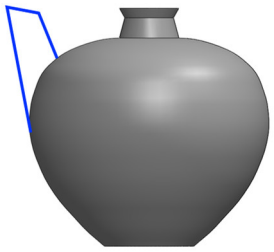
(*f*) Solid CAD model

Figure 4-14. Solid CAD modeling of the ewer body.

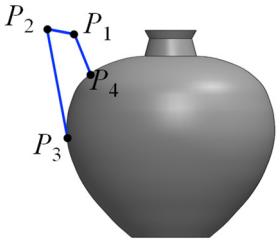
4.3.5. Solid CAD Modeling of Ewer Spout

Figure 4-15 shows the results of solid CAD modeling of the ewer spout. As shown in Figure 4-15(*b*), the spout's point cloud was injected into the ewer body's solid CAD model. Then the points P_1, P_2, P_3, and P_4 were defined, as shown in Figure 4-15(*c*). Here P_1 is the starting point of the first line segment, P_2 is the intersection point of the first and second line segment, P_3 is the intersection point of the second and third line segment, and P_4 is the endpoint of the third line segment of the spout. The segments are shown in Table 4-6. The point clouds must be iterated so that P_3 and P_4 must lie on a tangent to the surface of the body. Now, taking P_1 and P_2 as reference points, Plane-1 was constructed. Then ellipse was constructed on Plane-1 that connects points P_1 and P_2, as shown in Figure 4-15(*d*). Similarly, taking P_3 and P_4 as reference points, Plane-2 was constructed. Then another ellipse was constructed on Plane-2 that connects points P_3 and P_4, as shown in Figure 4-15(*d*). Next, two ellipses were joined by applying the lofted function, which results in the spout, as shown in Figure 4-15(*e*). Afterward, a hollow inside the solid CAD model was constructed by offsetting the surface and hiding the solid body. Finally, applying the thicken function on the offset surface results in a solid CAD model shown in Figure 4-15(*f*).

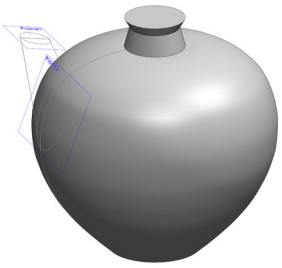
(*a*) Point cloud of the spout



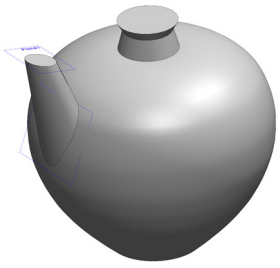
(*b*) Point cloud of the spout injected into the solid CAD model



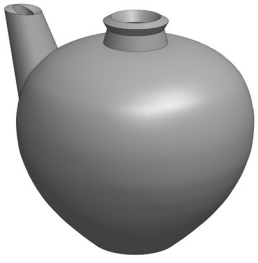
(*c*) Points are defined to construct planes



(*d*) Reference planes and ellipses to construct spout



(*e*) Top ellipse lofted to the bottom ellipse

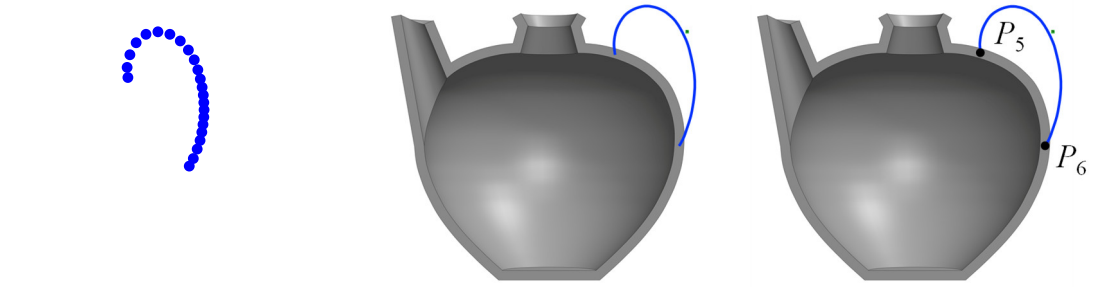


(*f*) A solid CAD model with hollow

Figure 4-15. Solid CAD modeling of the ewer spout.

4.3.6. Solid CAD modeling of Ewer Handle

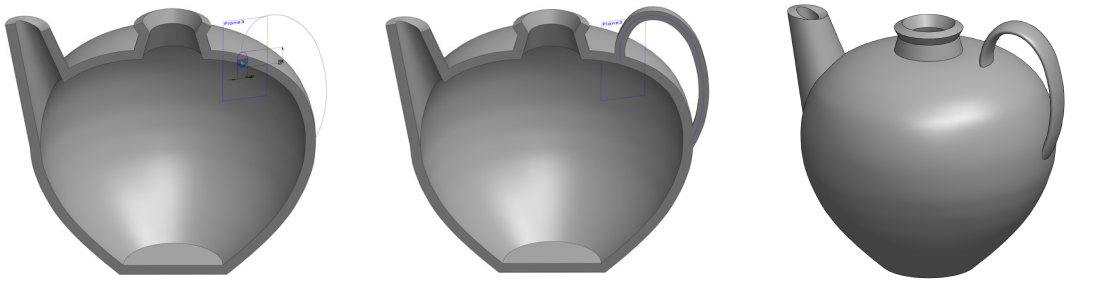
Figure 4-16 shows the results of solid CAD modeling of the ewer handle. As shown in Figure 4-16(*b*), the handle's point cloud was injected into the ewer body's solid CAD model. Then the points P_5 and P_6 were defined from the point cloud of the ewer handle. Here P_5 and P_6 are the starting and endpoint of the ewer handle, respectively. Iterate the handle's point cloud so that P_5 and P_6 must lie on the body's mid-section shown in Figure 4-16(*c*). Now, taking P_5 and handle segment as a reference, Plane-3 was constructed. Then the ellipse was constructed on Plane-3, as shown in Figure 4-16(*d*). The ellipse was then swept along the handle segment using the sweep function to construct the ewer handle, as shown in Figure 4-16(*e*). The result of the ewer's solid CAD modeling is shown in Figure 4-16(*f*).



(*a*) Point cloud of the handle

(*b*) Point cloud of the handle injected into the solid CAD model

(*c*) Convert to an entity and defined points to construct a plane



(*d*) Reference plane and ellipse to construct the handle

(*e*) The ellipse along the handle segment to construct the handle

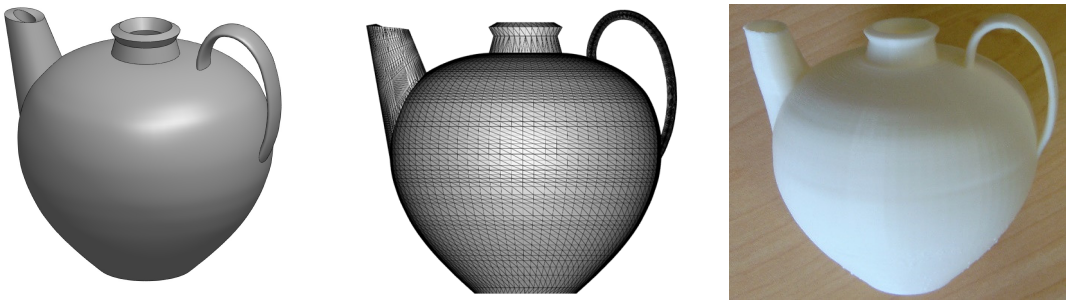
(*f*) Final solid CAD model of the ewer

Figure 4-16. Solid CAD modeling of the ewer handle.

4.3.7. 3D Printing of Ewer

After completing the solid CAD modeling, triangulation modeling was performed by just saving the solid CAD model into an STL file using a 3D CAD system. Figure

4-17(*a*) and Figure 4-17(*b*) show the ewer's solid CAD model (virtual model) and the triangulated model, respectively. The physical model was built from the STL data of the triangulated model using a 3D printer available at the Kitami Institute of Technology. The result of the 3D printed object is shown in Figure 4-17(*c*).



(*a*) Solid 3D CAD model (virtual model)

(*b*) Triangulated model

(*c*) 3D printed object (replica)

Figure 4-17. Ewer model.

The results shown in Figure 4-1 to Figure 4-17 demonstrated that the proposed reverse engineering can able to digitize Hokkaido Ainu motifs, patterns, and other significant cultural objects. Also, the proposed reverse engineering method can able to model both planner and complex shapes. The digital data of the virtual models are the important blueprint for manufacturing the physical objects (replicas). The next chapter addresses the discussion of this study.

Chapter 5: Discussion and Future Directions

This chapter addresses the full discussion, interpretation, and evaluation of the results of this study. The chapter first presents a general discussion (Section 5.1), followed by a specific discussion (Section 5.2), limitations (Section 5.3), and the future directions for this study (Section 5.4).

5.1. General Discussion

As stated in Chapter 1, the purpose of this study is to develop a novel reverse engineering method suitable for 3D printing without using a scanned point cloud, and the objectives are as follows:

1. develop a framework of a novel reverse engineering method;
2. develop an analytical point cloud creation process for the proposed reverse engineering method;
3. elucidate the efficacy of the proposed reverse engineering method by digitizing significant cultural objects; and
4. use 3D printing for prototyping.

The first and second objectives are elucidated in Chapter 3. The third and fourth objectives are elucidated in Chapter 4. All the questions emphasized in the introduction chapter are incorporated when creating a point cloud using the proposed reverse engineering method.

In this study, a database of ten basic motifs and six complex patterns belonging to Hokkaido Ainu are created. The database is made available to the public and can access online from the link shown in [151]. Besides digitizing basic motifs and complex patterns, their physical replicas are also built using an ordinary 3D printer. Moreover, the point cloud, virtual model, and physical model of an ancient ewer are created using the proposed reverse engineering method.

The study shows that the recursive process can create a point cloud of any desired shape. The point cloud points can be restricted to a small size and thus can be controlled

easily. As a result, the subsequent processes are easy to handle and free of complex computations. The point clouds are free from noise and outlier points. This way, the user has the freedom to design any given shape without being dependent on the data acquisition system and undergoing complex computation.

Moreover, the point clouds are used sequentially to produce the solid CAD model of the given object. It is not the case in conventional reverse engineering. It undergoes complex computations such as removing noise and outlier points, redundant points removal, and surface reconstruction from the point clouds. As a result, the process becomes the computational complexity and user-skill-dependent activity.

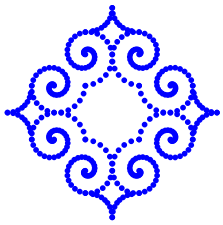
In the synopses, the outcomes of this study can help in two-folds. First, it will help those who want to digitize significant cultural objects without using a 3D scanner or image processing. Second, it will help those who want to simplify the current practices of reverse engineering. As such, the outcomes of this study can enrich the field of reverse engineering.

5.2. Specific Discussion

The recursive process implemented in this study is independent of the platform. As a result, analytical point clouds created by the recursive process are compatible with any software packages. For example, the analytical point cloud can be used for designing 2D images of motifs and patterns, which can be used for 2D printing devices. Such 2D images can be used for ornaments, walls, objects, and souvenir decorations. Moreover, the analytical point cloud can be used for surface texturing. Some insight into surface texturing is described as follows.

Let consider the solid CAD model of the ewer shown in Figure 4-17. A texture can be added to the solid CAD model. For example, consider the point clouds that represent a pattern taken from Hokkaido Ainu. The setting for creating the point cloud is shown in Figure 4-1. Figure 5-1(*a*) shows the point cloud of texturing. The point cloud is injected into the ewer's solid CAD model, as shown in Figure 5-1(*b*). Then, the texture on the front side is constructed by extruding the point cloud from the surface of the ewer. The texture on the backside is constructed by mirroring the front side texture. The result is shown in Figure 5-1(*c*). Finally, the physical model is realized using a 3D printer, as shown in

Figure 5-1(*d*). Thus, the analytical point clouds are useful for creating texture to enhance the beauty of the object.



(*a*) Point cloud

(*b*) Point cloud injected into the solid CAD model

(*c*) Textured solid CAD model

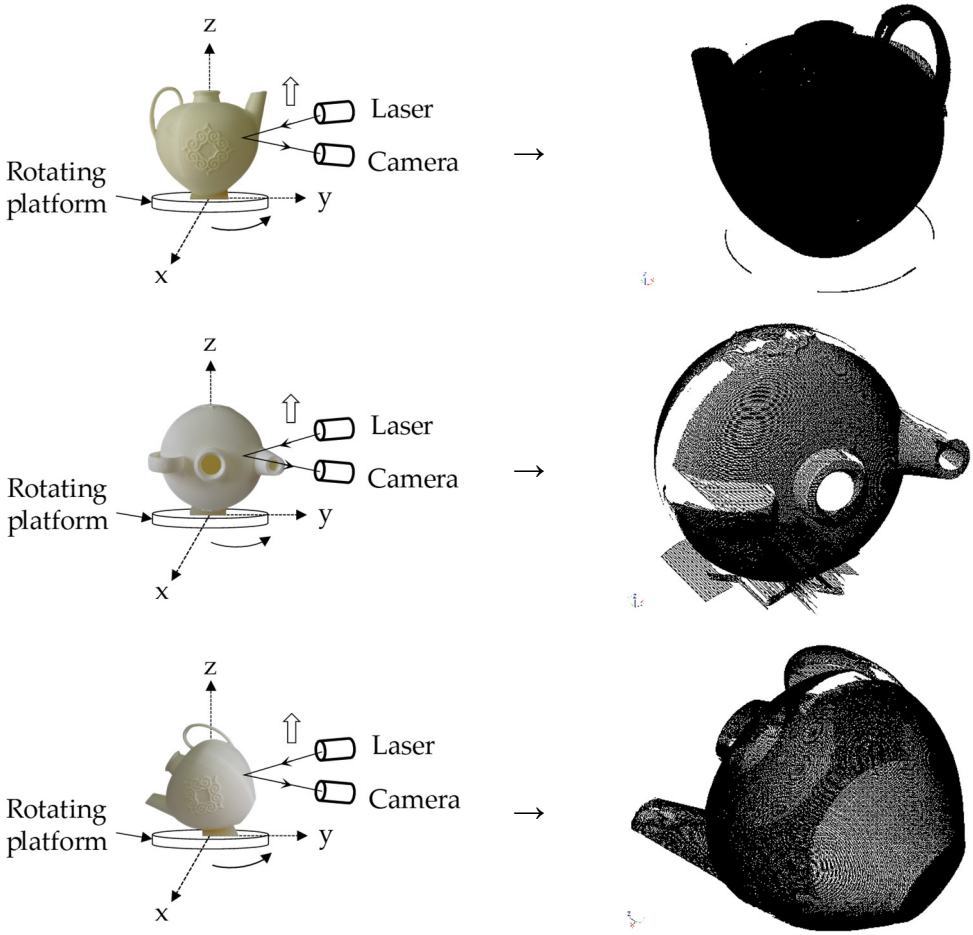
(*d*) 3D printed model

Figure 5-1. Surface texturing of the ewer.

Besides texturing, the 3D printed physical model of ewer shown in Figure 5-1(*d*) is scanned using the stationary Roland 3D scanner (model PICZA LPX-600) [30], which is available at the Kitami Institute of Technology. Figure 5-2 shows the point clouds obtained by scanning from three different orientations. Point clouds are then preprocessed (removing noise points, removing redundant points, combining all point clouds into a single coordinate) before creating a triangulated model. Afterward, the triangulation was performed. The result is shown in Figure 5-3. As seen in Figure 5-3, the texture is a blur, and there is also surface distortion near the handle. To make texture sharp, clear, and smooth, it requires the heavy user affords when performing solid CAD modeling.

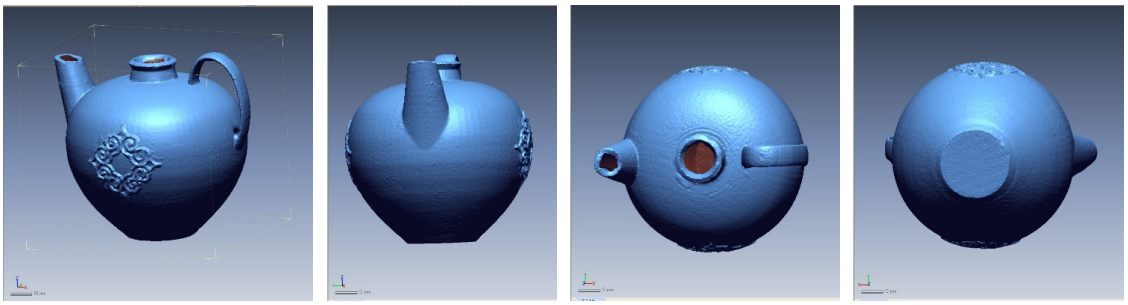
It is worth discussing that the triangulated model from the scanning-based modeling (Figure 5-3) consists of 271,803 facets and 815,409 vertices. In contrast, the triangulation model from the recursive process-based modeling has 25,336 facets and 76,008 vertices for the model without texture (Figure 4-17(*b*)) and 51,244 facets and 153,732 vertices for the model with texture (Figure 5-1(*d*)). Scanning-based modeling has ten times higher facets than recursive process-based modeling. As a result, scanning-based modeling

becomes user-skill-dependent and heavy-computation. Table 5-1 shows the comparison between the scanned-based modeling and the recursive process-based modeling.



(*a*) Scanning orientations (*b*) Point clouds

Figure 5-2. Point cloud registration from different orientations of scanning.



(*a*) Front view (*b*) Side view (*c*) Top view (*d*) Bottom view

Figure 5-3. Different views of a triangulated model obtained from the scanned-base point cloud datasets.

Table 5-1. Comparison between the scanned-based modeling and the recursive process-based modeling.

Parameters	Recursive Process-based Modeling		Scanning-based Modeling
	Model without Texture	Model with Texture	
Number of points	102	782	52,1404
Number of facets	25,336	51,244	271,803
Number of vertices	76,008	153,732	815,409
Printing time without support	7 h, 50 min, 38 s	8 h, 6 min, 58 s	-
Printing time with support	12 h, 20 min, 28 s	13 h, 14 min, 56 s	-

5.3. Limitations

No approach is without limitations, and the proposed reverse engineering method is not an exception. When the point clouds are created using the recursive process, the concentration of the points is higher when a radius of curvature is smaller. The concentration of the points is lower when the radius of curvature is large. For example, see Figure 3-12. It has the same issue when the point clouds are created using parametric-based equations (e.g., Bézier curve).

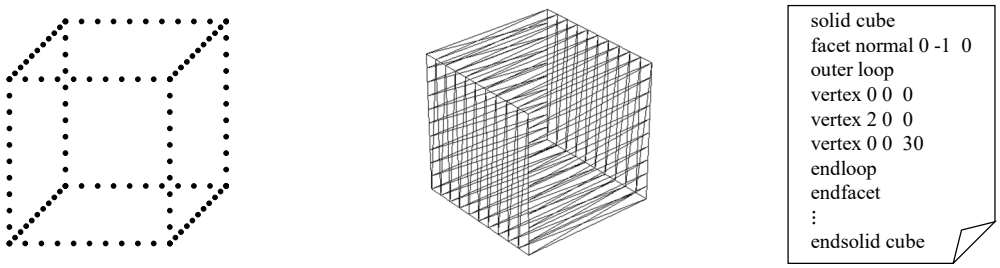
Interestingly, the electric charge also exhibits similar characteristics [152]. There is a scope for further investigation of this phenomenon. However, this study did not investigate this phenomenon because the precision of the model created by the proposed reverse engineering method is satisfactory to model the given objects.

In the proposed reverse engineering method, the interpretation of the shape outline relies on human perception only. Based on this shape outline, the recursive process creates the point cloud. In this respect, other aiding technology, e.g., 3D pen-based shape learning technology [31], could be incorporated to enrich the presented methodology. These issues remain open for further research.

5.4. Future Directions

In the future, this study may be extended in two directions. In the first direction, the analytical point cloud can be directly triangulated by eliminating solid CAD modeling.

The STL data of the triangulated model can be used to manufacture the physical object. An example is shown in Figure 5-4. This way, reverse engineering can be achieved without using sophisticated devices such as a 3D scanner and 3D CAD system.



(*a*) Analytical point cloud (*b*) Triangluated model (*c*) STL data

Figure 5-4. Example of direct triangulation of an analytical point cloud of a cube and its STL dataset.

In the second direction, this study can be extended to develop a system that automatically generates a shape based on the linguistic expression defined by the user. The concept of human-cognition-based reverse engineering is illustrated in Figure 5-5. This way, human cognition-based reverse engineering can be achieved, which is one of the frontiers of digital manufacturing.

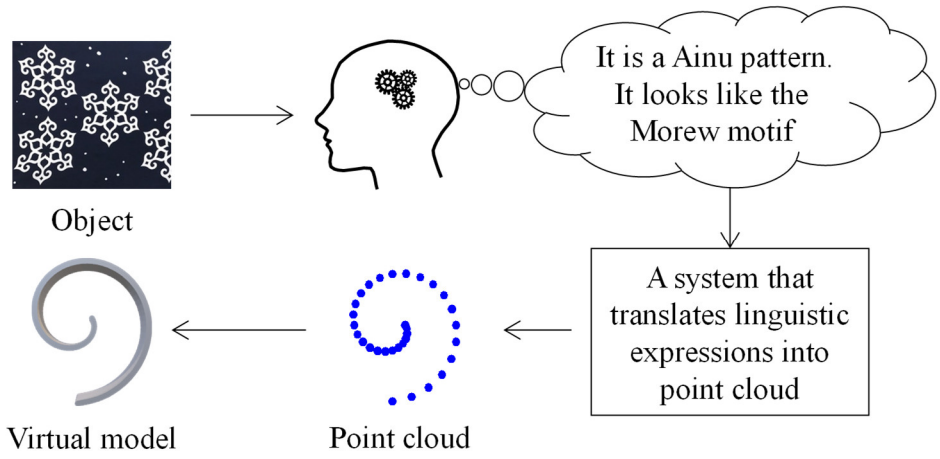


Figure 5-5. Concept of human-cognition-based reverse engineering.

There are other open issues for future research. For example, what if one superimposes a 3D scanner-based point cloud on the point cloud obtained by the proposed method? If the point cloud obtained by the proposed method is considered the ideal point cloud, then the point cloud obtained by a 3D scanner can be processed based on the ideal one. It might be open and a new direction for point cloud-based reverse engineering. The next chapter addresses the conclusion of this study.

Chapter 6: Conclusions

In this study, a novel reverse engineering method suitable for 3D printing and its application to preserving cultural heritage has been presented. In the proposed reverse engineering method, a point cloud of the desired shape is created by the recursive process.

The recursive process creates an analytical point cloud using two variable inputs and three constant inputs. The two variable inputs are denoted as instantaneous distance and instantaneous angle, which vary in each iteration of the recursive process. The constant inputs are denoted as the center point, initial distance, and initial angle, which remain constant for all iterations. It is found that by varying pairs of the instantaneous rotational angle and the instantaneous distance systematically, the recursive process can able to model both planner and complex shapes. In most cases, the instantaneous rotational angle is kept simple (i.e., constant or linear increase or linear decrease) and the instantaneous distance is varied using simple functions (e.g., straight-line, tent function, sine/cosine, u-shape function, and bathtub function). For example, a well-known shape such as a straight-line, circle, ellipses, spiral, astroid, S-shape, and leaf-shape are modeled using the recursive process.

On the other hand, three constant inputs also play vital roles. The center point can perform the roles of translation function in both the x-axis and y-axis directions. The initial angle can perform the roles of the rotation function on the x-y plane. The third constant does not play any roles, however, it is necessary to create the starting point.

The modeling ability of the proposed reverse engineering method is validated by comparing two separate point clouds of the same shape. The first point cloud is created using the parametric equation for a given shape (e.g., circle, ellipse, spiral, astroid, and straight lines). The second point cloud of the same shape is created using the recursive process. It is observed that the point clouds of both shapes look identical. Further, the point clouds are quantified by the parameter called the radius of curvature and aesthetic value. It is observed that both shapes have the same radius of curvature and aesthetic value. Hence, even though the shapes are modeled using the recursive process instead of the respective parametric equation-based, their aesthetic natures remain unaffected. It is

also found that the instantaneous distance plays the role of a radius of curvature. So, it is not difficult to predict the instantaneous distance values than those of the curvature radius while modeling an arbitrary shape.

The efficacy of the proposed reverse engineering method is demonstrated using two applications. In the first application, a database of ten Hokkaido Ainu motifs is created. The point clouds, virtual models, and physical models of ten Hokkaido Ainu motifs and six complex patterns are realized using the proposed reverse engineering method. In the second application, the point cloud, virtual model, and physical model of an ancient ewer having a 3D shape are realized using the proposed reverse engineering method. All physical models are built using an ordinary 3D printer available at the Kitami Institute of Technology. Setting the input parameters to create the respective point clouds using the recursive process was also described elaborately.

The results show that the proposed reverse engineering method avoids sophisticated and complex computations, which are unavoidable in conventional reverse engineering. It also makes reverse engineering more systematic to model complex shapes. The proposed reverse engineering method also eliminated the use of a 3D scanner and its scanned point cloud. In addition to the shape modeling, the point cloud created by the recursive process can be used for surface texturing. Therefore, one can use the proposed reverse engineering method for creating aesthetic artifacts lucidly to have complex geometry.

Bibliography

1. Zhaohui Geng and Bopaya Bidanda. Review of reverse engineering systems – current state of the art. *Virtual and Physical Prototyping*, 2017, Vol. 12, No. 2, pp. 161-172. DOI: 10.1080/17452759.2017.1302787.
2. Alexandru C. Telea. Reverse Engineering - Recent Advances and Applications. 1st ed. 2012, InTech.
3. Wego Wang, *Reverse Engineering: Technology of Reinvention*, 1st ed., CRC Press: Boca Raton, Florida, USA, 2010.
4. Morteza Daneshmand, Ahmed Helmi, Egils Avots, Fatemeh Noroozi, Fatih Alisinanoglu, Hasan Sait Arslan, Jelena Gorbova, Rain Eric Haamer, Cagri Ozcinar, and Gholamreza Anbarjafari. 3D Scanning: A Comprehensive Survey. *Computer Vision and Pattern Recognition*, 2018, pp. 1-18. DOI: arXiv:1801.08863.
5. David W. James, Fawzi Belblidia, Jurgen E. Eckermann, and Johann Sienz. An innovative photogrammetry color segmentation based technique as an alternative approach to 3D scanning for reverse engineering design. *Computer-Aided Design and Applications*, 2017, Vol. 14, No. 1, pp. 1-16. DOI: 10.1080/16864360.2016.1199751.
6. Eldad Eilam, *Reversing: Secrets of Reverse Engineering*, 1st ed., Wiley: Indianapolis, Indiana, 2005.
7. Tamás Várady, Ralph R. Martin, and Jordan Cox. Reverse engineering of geometric models—an introduction. *Computer-Aided Design*, 1997, Vol. 29, No. 4, pp. 255-268. DOI: 10.1016/S0010-4485(96)00054-1.
8. Wego Wang. Application of Reverse Engineering in Manufacturing Industry. In *Eastec 2013 - Society of Manufacturing Engineers*, West Springfield, MA, USA, 2013, pp. 1-7.
9. Chi Hieu Le, Nikolay Zlatov, Jos Vander Sloten, J. Vander Sloten, Erik Bohez, L. Khanh, P.H. Binh, P. Oris, and Yuli Toshev. Medical rapid prototyping applications and methods. *Assembly Automation*, 2005, Vol. 25, No. 4, pp. 284-292. DOI: 10.1108/01445150510626415.

10. Jean Loup Faulon, W. Michael Brown, and Shawn Martin. Reverse engineering chemical structures from molecular descriptors: how many solutions? *Journal of Computer-Aided Molecular Design*, 2005, Vol. 19, No. 9, pp. 637-650. DOI: 10.1007/s10822-005-9007-1.

11. Hiroshi Masuda, Takeru Niwa, Ichiro Tanaka, and Ryo Matsuoka. Reconstruction of Polygonal Faces from Large-Scale Point-Clouds of Engineering Plants. *Computer-Aided Design and Applications*, 2015, Vol. 12, No. 5, pp. 555-563. DOI: 10.1080/16864360.2015.1014733.

12. Randy Torrance and Dick James. *The State-of-the-Art in IC Reverse Engineering*, in *Cryptographic Hardware and Embedded Systems - CHES 2009*, Vol. 5747, Christophe Clavier and Kris Gaj, Editors. 2009, Springer Berlin Heidelberg: Berlin, Heidelberg. pp. 363-381. DOI: 10.1007/978-3-642-04138-9_26.

13. Nabil Anwer and Luc Mathieu. From reverse engineering to shape engineering in mechanical design. *CIRP Annals*, 2016, Vol. 65, No. 1, pp. 165-168. DOI: 10.1016/j.cirp.2016.04.052.

14. Tashi and AMM Sharif Ullah. Symmetrical Patterns of Ainu Heritage and Their Virtual and Physical Prototyping. *Symmetry*, 2019, Vol. 11, No. 8, pp. 985(981)-985(914). DOI: 10.3390/sym11080985.

15. Sofia Ceccarelli, Massimiliano Guarneri, Mario Ferri de Collibus, Massimo Francucci, Massimiliano Ciaffi, and Alessandro Danielis. Laser Scanners for High-Quality 3D and IR Imaging in Cultural Heritage Monitoring and Documentation. *Journal of Imaging*, 2018, Vol. 4, pp. 4(11). DOI: 10.3390/jimaging4110130.

16. Chang Kuang-Hua. *A Review on Shape Engineering and Design Parameterization in Reverse Engineering*, Alexandru C. Telea, Editor. 2012, INTECH Open Access Publisher. DOI: 10.5772/32419.

17. Francesco Buonamici, Monica Carfagni, Rocco Furferi, Lapo Governi, Alessandro Lapini, and Yary Volpe. Reverse engineering of mechanical parts: A template-based approach. *Journal of Computational Design and Engineering*, 2018, Vol. 5, No. 2, pp. 145-159. DOI: 10.1016/j.jcde.2017.11.009.

18. Dinesh K. Pal, Laxman S. Bhargava, B. Ravi, and U. Chandrasekhar. Computer-aided Reverse Engineering for Rapid Replacement of Parts. *Defence Science Journal*, 2006, Vol. 56, No. 2, pp. 225-238. DOI: 10.14429/dsj.56.1885.

19. Ian Gibson, David Rosen, and Brent Stucker, Additive Manufacturing Technologies: 3D Printing, Rapid Prototyping, and Direct Digital Manufacturing, 2nd ed., Springer-Verlag: New York, 2015.

20. Wei Gao, Yunbo Zhang, Devarajan Ramanujan, Karthik Ramani, Yong Chen, Christopher B. Williams, Charlie C. L. Wang, Yung C. Shin, Song Zhang, and Pablo D. Zavattieri. The status, challenges, and future of additive manufacturing in engineering. *Computer-Aided Design*, 2015, Vol. 69, pp. 65-89. DOI: 10.1016/j.cad.2015.04.001.

21. Syed A. M. Tofail, Elias P. Koumoulos, Amit Bandyopadhyay, Susmita Bose, Lisa O'Donoghue, and Costas Charitidis. Additive manufacturing: scientific and technological challenges, market uptake and opportunities. *Materials Today*, 2018, Vol. 21, No. 1, pp. 22-37. DOI: 10.1016/j.mattod.2017.07.001.

22. Caterina Balletti, Martina Ballarin, and Francesco Guerra. 3D printing: State of the art and future perspectives. *Journal of Cultural Heritage*, 2017, Vol. 26, No. Supplement C, pp. 172-182. DOI: 10.1016/j.culher.2017.02.010.

23. Rafiq Nooran, *3D Printing: Technology, Applications, and Selection*, 1st ed., CRC Press: Boca Raton, Florida, USA, 2017.

24. Basilio Ramos Barbero and Elena Santos Ureta. Comparative study of different digitization techniques and their accuracy. *Computer-Aided Design*, 2011, Vol. 43, No. 2, pp. 188-206. DOI: 10.1016/j.cad.2010.11.005.

25. Igor Budak, Djordje Vukelić, Drago Bračun, Janko Hodolič, and Mirko Soković. Pre-Processing of Point-Data from Contact and Optical 3D Digitization Sensors. *Sensors*, 2012, Vol. 12, No. 1, pp. 1100-1126. DOI: 10.3390/s120101100.

26. F. Smith. *Point-based Geometric Modelling*, in *Machining Impossible Shapes*, Gustav J. Olling, Byoung K. Choi, and Robert B. Jerard, Editors. 1999, Springer US: Boston, MA. pp. 74-81. DOI: 10.1007/978-0-387-35392-0_7.

27. Bill Toll and Fuhua Cheng. *Surface Reconstruction from Point Clouds*, in *Machining Impossible Shapes*, Gustav J. Olling, Byoung K. Choi, and Robert B.

Jerard, Editors. 1999, Springer US: Boston, MA. pp. 173-178. DOI: 10.1007/978-0-387-35392-0_18.

28. Tamás Várady, Pál Benkő, Géza Kós, Gábor Renner, and Volker Weiß. *Segmentation and Surface Fitting in Reverse Engineering*, in *Machining Impossible Shapes*, Gustav J. Olling, Byoung K. Choi, and Robert B. Jerard, Editors. 1999, Springer US: Boston, MA. pp. 167-172. DOI: 10.1007/978-0-387-35392-0_17.

29. Vera B. Anand, *Computer Graphics and Geometric Modeling for Engineers*, 1st ed., John Wiley & Sons: New York, United States, 1996.

30. Roland 3D Laser Scanner. Available online: https://www.rolanddga.com/support/products/3d-scanners/picza-lpx-600re-3d-laser-scanner (accessed on June 8, 2020)

31. Hammad Mian Syed, Abdul Mannan Mohammed, and M. Al-Ahmari Abdulrahman. The influence of surface topology on the quality of the point cloud data acquired with laser line scanning probe. *Sensor Review*, 2014, Vol. 34, No. 3, pp. 255-265. DOI: 10.1108/SR-01-2013-611.

32. Tashi, AMM Sharif Ullah, Michiko Watanabe, and Akihiko Kubo. Analytical Point-Cloud Based Geometric Modeling for Additive Manufacturing and Its Application to Cultural Heritage Preservation. *Applied Sciences*, 2018, Vol. 8, No. 5, pp. 656(651)-656(618). DOI: 10.3390/app8050656.

33. AMM Sharif Ullah, Ryuta Omori, Yuji Nagara, Akihiko Kubo, and Jun'ichi Tamaki. Toward Error-free Manufacturing of Fractals. *Procedia CIRP*, 2013, Vol. 12, No. Supplement C, pp. 43-48. DOI: 10.1016/j.procir.2013.09.009.

34. AMM Sharif Ullah, Yoshimi Sato, Akihiko Kubo, and Jun'ichi Tamaki. Design for Manufacturing of IFS Fractals from the Perspective of Barnsley's Fern-leaf. *Computer-Aided Design and Applications*, 2015, Vol. 12, No. 3, pp. 241-255. DOI: 10.1080/16864360.2014.981452.

35. AMM Sharif Ullah, Doriana Marilena D'Addona, Khalifa H. Harib, and Than Kin. Fractals and Additive Manufacturing. *International journal of automation technology*, 2016, Vol. 10, No. 2, pp. 222-230. DOI: 10.20965/ijat.2016.p0222.

36. AMM Sharif Ullah. Design for additive manufacturing of porous structures using stochastic point-cloud: a pragmatic approach. *Computer-Aided Design and*

Applications, 2018, Vol. 15, No. 1, pp. 138-146. DOI: 10.1080/16864360.2017.1353747.

37. M. G. Rekoff. On reverse engineering. *IEEE Transactions on Systems, Man, and Cybernetics*, 1985, Vol. SMC-15, No. 2, pp. 244-252. DOI: 10.1109/TSMC.1985.6313354.

38. Elliot J. Chikofsky and James H. Cross. Reverse engineering and design recovery: a taxonomy. *IEEE Software*, 1990, Vol. 7, No. 1, pp. 13-17. DOI: 10.1109/52.43044.

39. Kevin N. Otto and Kristin L. Wood. Product Evolution: A Reverse Engineering and Redesign Methodology. *Research in Engineering Design*, 1998, Vol. 10, No. 4, pp. 226-243. DOI: 10.1007/s001639870003.

40. IEEE Standard for Software Maintenance. *IEEE Std 1219-1993*, 1993, pp. 1-45.

41. Philip. H. Francis. Project Management, in *In Tool and Manufacturing Engineers Handbook*, Vol. 5: Manufacturing Management, Raymond. F. Veilleux and Louis. W. Petro, Editors. 1988, Society of Manufacturing Engineers: Dearborn. pp. 17-20.

42. ASTM International. *Standard Terminology for Additive Manufacturing Technologies, (Withdrawn 2015)*. ASTM International, F2792-12a, West Conshohocken, PA, 2012, DOI: 10.1520/F2792-12A.

43. Francesco Buonamici and Monica Carfagni. Reverse Engineering of Mechanical Parts: A Brief Overview of Existing Approaches and Possible New Strategies. In *ASME 2016 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, August 21–24, 2016, Charlotte, North Carolina, USA. DOI: 10.1115/detc2016-59242.

44. Zhonggui Chen, Wen Chen, Jianzhi Guo, Juan Cao, and Yongjie Jessica Zhang. Orientation field guided line abstraction for 3D printing. *Computer Aided Geometric Design*, 2018, Vol. 62, pp. 253-262. DOI: 10.1016/j.cagd.2018.03.014.

45. Ariel Schwartz, Ronit Schneor, Gila Molcho, and Miri Weiss Cohen. Surface detection and modeling of an arbitrary point cloud from 3D sketching. *Computer-Aided Design and Applications*, 2017, pp. 1-11. DOI: 10.1080/16864360.2017.1375673.

46. Pedro Company, Manuel Contero, Julian Conesa, and Ana Piquer. An optimisation-based reconstruction engine for 3D modelling by sketching. *Computers & Graphics*, 2004, Vol. 28, No. 6, pp. 955-979. DOI: 10.1016/j.cag.2004.08.007.

47. Ferran Naya, Julián Conesa, Manuel Contero, Pedro Company, and Joaquim Jorge. Smart Sketch System for 3D Reconstruction Based Modeling. In *Third International Symposium on Smart Graphics*, Berlin/Heidelberg, Germany, 2003, pp. 58-68. DOI: 10.1007/11555261_11.

48. Jun'ichi Tamaki, Satoshi Yanagi, Yuya Aoki, Akihiko Kubo, Takao Kameda, and AMM Sharif Ullah. 3D Reproduction of a Snow Crystal by Stereolithography. *Journal of Advanced Mechanical Design, Systems, and Manufacturing*, 2012, Vol. 6, No. 6, pp. 923-935. DOI: 10.1299/jamdsm.6.923.

49. José Ignacio Rojas-Sola, Belén Galán-Moral, and Eduardo De la Morena-De la Fuente. Agustín de Betancourt's Double-Acting Steam Engine: Geometric Modeling and Virtual Reconstruction. *Symmetry*, 2018, Vol. 10, No. 8, pp. 351(351)-351(319). DOI: 10.3390/sym10080351.

50. Rocco Furferi, Lapo Governi, Yary Volpe, Luca Puggelli, Niccolò Vanni, and Monica Carfagni. From 2D to 2.5D i.e. from painting to tactile model. *Graphical Models*, 2014, Vol. 76, No. 6, pp. 706-723. DOI: 10.1016/j.gmod.2014.10.001.

51. AMM Sharif Ullah, Hiroki Kiuno, Akihiko Kubo, and Doriana Marilena D'Addona. A system for designing and 3D printing of porous structures. *CIRP Annals*, 2020, Vol. 69, No. 1, pp. 113-116. DOI: 10.1016/j.cirp.2020.04.088.

52. Qingjin Peng and Hector Sanchez. 3D Digitizing Technology in Product Reverse Design. In *Proceedings of the Canadian Design Engineering Network (CDEN)*, Kaninaskis, Alberta, July 18-20, 2005, pp. 1-10. DOI: 10.24908/pceea.v0i0.3966.

53. Jiaqi Yang, Zhiguo Cao, and Qian Zhang. A fast and robust local descriptor for 3D point cloud registration. *Information Sciences*, 2016, Vol. 346-347, No. Supplement C, pp. 163-179. DOI: 10.1016/j.ins.2016.01.095.

54. Weimin Li and Pengfei Song. A modified ICP algorithm based on dynamic adjustment factor for registration of point cloud and CAD model. *Pattern Recognition Letters*, 2015, Vol. 65, No. Supplement C, pp. 88-94. DOI: 10.1016/j.patrec.2015.07.019.

55. Xiaojun Zhao and Horea T. Ilieş. Learned 3D shape descriptors for classifying 3D point cloud models. *Computer-Aided Design and Applications*, 2017, Vol. 14, No. 4, pp. 507-515. DOI: 10.1080/16864360.2016.1257192.

56. Sukwoo Jung, Seunghyun Song, Minho Chang, and Sangchul Park. Range image registration based on 2D synthetic images. *Computer-Aided Design*, 2018, Vol. 94, pp. 16-27. DOI: 10.1016/j.cad.2017.08.001.

57. Yap Yee Ling, Tan Yong Sheng Edgar, Tan Heang Kuan Joel, Peh Zhen Kai, Low Xue Yi, Yeong Wai Yee, Tan Colin Siang Hui, and Laude Augustinus. 3D printed bio-models for medical applications. *Rapid Prototyping Journal*, 2017, Vol. 23, No. 2, pp. 227-235. DOI: 10.1108/RPJ-08-2015-0102.

58. Zhonggui Chen, Tieyi Zhang, Juan Cao, Yongjie Jessica Zhang, and Cheng Wang. Point cloud resampling using centroidal Voronoi tessellation methods. *Computer-Aided Design*, 2018, Vol. 102, pp. 12-21. DOI: 10.1016/j.cad.2018.04.010.

59. Martin Skrodzki, Johanna Jansen, and Konrad Polthier. Directional density measure to intrinsically estimate and counteract non-uniformity in point clouds. *Computer Aided Geometric Design*, 2018, Vol. 64, pp. 73-89. DOI: 10.1016/j.cagd.2018.03.011.

60. Kwan H. Lee and H. Woo. Direct integration of reverse engineering and rapid prototyping. *Computers & Industrial Engineering*, 2000, Vol. 38, No. 1, pp. 21-38. DOI: 10.1016/S0360-8352(00)00017-6.

61. Truc Le and Ye Duan. A primitive-based 3D segmentation algorithm for mechanical CAD models. *Computer Aided Geometric Design*, 2017, Vol. 52-53, pp. 231-246. DOI: 10.1016/j.cagd.2017.02.009.

62. Reed M. Williams and Horea T. Ilieş. Practical shape analysis and segmentation methods for point cloud models. *Computer Aided Geometric Design*, 2018, Vol. 67, pp. 97-120. DOI: 10.1016/j.cagd.2018.10.003.

63. Pierre-Alain Fayolle and Alexander Pasko. An evolutionary approach to the extraction of object construction trees from 3D point clouds. *Computer-Aided Design*, 2016, Vol. 74, pp. 1-17. DOI: 10.1016/j.cad.2016.01.001.

64. Maodong Pan, Weihua Tong, and Falai Chen. Phase-field guided surface reconstruction based on implicit hierarchical B-splines. *Computer Aided Geometric Design*, 2017, Vol. 52-53, pp. 154-169. DOI: 10.1016/j.cagd.2017.03.009.

65. Chen Feng and Yuichi Taguchi. FasTFit: A fast T-spline fitting algorithm. *Computer-Aided Design*, 2017, Vol. 92, pp. 11-21. DOI: 10.1016/j.cad.2017.07.002.

66. Jinting Xu, Wenbin Hou, and Hongzhe Zhang. An improved virtual edge approach to slicing of point cloud for additive manufacturing. *Computer-Aided Design and Applications*, 2017, pp. 1-7. DOI: 10.1080/16864360.2017.1397884.

67. Huang Haiqiao, Mok P.Y., Kwok Y.L., and Au J.S. Automatic Block Pattern Generation from a 3D Unstructured Point Cloud. *Research Journal of Textile and Apparel*, 2010, Vol. 14, No. 1, pp. 26-37. DOI: 10.1108/RJTA-14-01-2010-B003.

68. Martin Peternell and Tibor Steiner. Reconstruction of piecewise planar objects from point clouds. *Computer-Aided Design*, 2004, Vol. 36, No. 4, pp. 333-342. DOI: 10.1016/S0010-4485(03)00102-7.

69. Pal Pralay. An easy rapid prototyping technique with point cloud data. *Rapid Prototyping Journal*, 2001, Vol. 7, No. 2, pp. 82-90. DOI: 10.1108/13552540110386709.

70. Nathan Litke Caltech, Adi Levin, and Peter Schroder Caltech. Fitting subdivision surfaces. In *VIS01: IEEE Visualization 2001*, San Diego, CA, USA, October 21-26, 2001, pp. 319-568. DOI: 10.1109/VISUAL.2001.964527.

71. Ji Ma, Jack Szu-Shen Chen, Hsi-Yung Feng, and Lihui Wang. Automatic construction of watertight manifold triangle meshes from scanned point clouds using matched umbrella facets. *Computer-Aided Design and Applications*, 2017, Vol. 14, No. 6, pp. 742-750. DOI: 10.1080/16864360.2017.1287676.

72. Shan Zhong, Yongqiang Yang, and Yanlu Huang. Data Slicing Processing Method for RE/RP System Based on Spatial Point Cloud Data. *Computer-Aided Design and Applications*, 2014, Vol. 11, No. 1, pp. 20-31. DOI: 10.1080/16864360.2013.834133.

73. Ming Ching Chang, Frederic Fol Leymarie, and Benjamin B. Kimia. Surface reconstruction from point clouds by transforming the medial scaffold. *Computer*

Vision and Image Understanding, 2009, Vol. 113, No. 11, pp. 1130-1146. DOI: 10.1016/j.cviu.2009.04.001.

74. Oussama Remil, Qian Xie, Xingyu Xie, Kai Xu, and Jun Wang. Surface reconstruction with data-driven exemplar priors. *Computer-Aided Design*, 2017, Vol. 88, pp. 31-41. DOI: 10.1016/j.cad.2017.04.004.

75. Mary Kathryn Thompson, Giovanni Moroni, Tom Vaneker, Georges Fadel, R. Ian Campbell, Ian Gibson, Alain Bernard, Joachim Schulz, Patricia Graf, Bhrigu Ahuja, and Filomeno Martina. Design for Additive Manufacturing: Trends, opportunities, considerations, and constraints. *CIRP Annals*, 2016, Vol. 65, No. 2, pp. 737-760. DOI: 10.1016/j.cirp.2016.05.004.

76. Gianluca Percoco and Luigi Maria Galantucci. Local-genetic slicing of point clouds for rapid prototyping. *Rapid Prototyping Journal*, 2008, Vol. 14, No. 3, pp. 161-166. DOI: 10.1108/13552540810878021.

77. Xiaochao Wang, Jianping Hu, Lixin Guo, Dongbo Zhang, Hong Qin, and Aimin Hao. Feature-preserving, mesh-free empirical mode decomposition for point clouds and its applications. *Computer Aided Geometric Design*, 2018, Vol. 59, pp. 1-16. DOI: 10.1016/j.cagd.2017.11.002.

78. Pinghai Yang, Tim Schmidt, and Xiaoping Qian. Direct Digital Design and Manufacturing from Massive Point-Cloud Data. *Computer-Aided Design and Applications*, 2009, Vol. 6, No. 5, pp. 685-699. DOI: 10.3722/cadaps.2009.685-699.

79. William Oropallo, Les A. Piegl, Paul Rosen, and Khairan Rajab. Generating point clouds for slicing free-form objects for 3-D printing. *Computer-Aided Design and Applications*, 2017, Vol. 14, No. 2, pp. 242-249. DOI: 10.1080/16864360.2016.1223443.

80. William Oropallo, Les A. Piegl, Paul Rosen, and Khairan Rajab. Point cloud slicing for 3-D printing. *Computer-Aided Design and Applications*, 2018, Vol. 15, No. 1, pp. 90-97. DOI: 10.1080/16864360.2017.1353732.

81. Roberto Scopigno, Paolo Cignoni, Nico Pietroni, Marco Callieri, and Matteo Dellepiane. Digital Fabrication Techniques for Cultural Heritage: A Survey.

Computer Graphics Forum, 2017, Vol. 36, No. 1, pp. 6-21. DOI: 10.1111/cgf.12781.

82. Ruggero Pintus, Kazim Pal, Ying Yang, Tim Weyrich, Enrico Gobbetti, and Holly Rushmeier. A Survey of Geometric Analysis in Cultural Heritage. 2016, Vol. 35, No. 1, pp. 4-31. DOI: 10.1111/cgf.12668.

83. Min-Bin Chen, Ya-Ning Yen, Hung-Ming Cheng, and Wun-Bin Yang. An Application of Reverse Engineering to the Digitization of Cultural Heritage Building, in Recent Advances in Computer Science and Information Engineering: Volume 3, Zhihong Qian, Lei Cao, Weilian Su, Tingkai Wang, and Huamin Yang, Editors. 2012, Springer Berlin Heidelberg: Berlin, Heidelberg. pp. 391-397. DOI: 10.1007/978-3-642-25766-7_52.

84. Mona Hess and Stuart Robson. Re-engineering Watt: A case study and best practice recommendations for 3D colour laser scans and 3D printing in museum artefact documentation. In *Lacona IX - Lasers in conservation*, British Museum, London, 2013, pp. 154-162.

85. Reichinger A. Neumüller M., Rist F., Kern C. 3D Printing for Cultural Heritage: Preservation, Accessibility, Research and Education, Ioannides M., Quak E., (Eds.). *3D Research Challenges in Cultural Heritage*, 2014, Vol. 8355, pp. 119-134. DOI: 10.1007/978-3-662-44630-0_9.

86. Kyoko Hasegawa, Liang Li, Naoya Okamoto, Shu Yanai, Hiroshi Yamaguchi, Atsushi Okamoto, and Satoshi Tanaka. Application of Stochastic Point-Based Rendering to Laser-Scanned Point Clouds of Various Cultural Heritage Objects. *IJAT*, 2018, Vol. 12, No. 3, pp. 348-355.

87. X. J. Cheng and W. Jin. Study on Reverse Engineering of Historical Architecture Based on 3D Laser Scanner. *Journal of Physics: Conference Series*, 2006, Vol. 48, pp. 843-849. DOI: 10.1088/1742-6596/48/1/160.

88. Young Hoon Jo and Seonghyuk Hong. Three-Dimensional Digital Documentation of Cultural Heritage Site Based on the Convergence of Terrestrial Laser Scanning and Unmanned Aerial Vehicle Photogrammetry. *ISPRS International Journal of Geo-Information*, 2019, Vol. 8, No. 2, pp. 53. DOI: 10.3390/ijgi8020053.

89. Emanuel M. Sachs, Michael J. Cima, and James A. Cornie. Three-Dimensional Printing: Rapid Tooling and Prototypes Directly from a CAD Model. *CIRP Annals*, 1990, Vol. 39, No. 1, pp. 201-204. DOI: 10.1016/S0007-8506(07)61035-X.

90. David L. Bourella, Joseph J. Beaman, Ming C. Leub, and David W. Rosen. A Brief History of Additive Manufacturing and the 2009 Roadmap for Additive Manufacturing: Looking Back and Looking Ahead. *US – TURKEY Workshop On Rapid Technologies*, 2009, pp. 5-11.

91. Hideo Kodama. Automatic method for fabricating a three-dimensional plastic model with photo-hardening polymer. *Review of Scientific Instruments*, 1981, Vol. 52, No. 11, pp. 1770-1773. DOI: 10.1063/1.1136492.

92. Our Story. Available online: https://www.3dsystems.com/our-story (accessed on October 2020)

93. David L. Bourell, David W. Rosen, and Ming C. Leu. The Roadmap for Additive Manufacturing and Its Impact. 2014, Vol. 1, No. 1, pp. 6-9. DOI: 10.1089/3dp.2013.0002.

94. Hype Cycle for 3D Printing, 2019. Available online: https://www.gartner.com/en/documents/3947508 (accessed on June 2, 2020)

95. Wohlers Report 2020. Available online: http://wohlersassociates.com (accessed on October 20, 2020)

96. 3D printing trends 2020. Available online: https://www.3dhubs.com/guides/3d-printing/ (accessed on June 2, 2020)

97. The State of 3D Printing. Available online: https://cdn2.hubspot.net/hubfs/5154612/downloads/Sculpteo_The%20State%20of%203D%20Printing_2019.pdf (accessed on June 2, 2020)

98. Phil Reeves and Dinusha Mendis. The Current Status and Impact of 3D Prinitng within the Industrial Sector: An Analysis of Six Case Studies. 2015, Bournemouth University.

99. Mathieu Terner. The Current State, Outcome and Vision of Additive Manufacturing. *Journal of Welding and Joining*, 2015, Vol. 33, No. 6, pp. 1-5. DOI: 10.5781/JWJ.2015.33.6.1.

100. Mohd Javaid and Abid Haleem. Current status and challenges of Additive manufacturing in orthopaedics: An overview. *Journal of Clinical Orthopaedics and Trauma*, 2019, Vol. 10, No. 2, pp. 380-386. DOI: 10.1016/j.jcot.2018.05.008.

101. International Data Corporation (IDC) Spending Guide. Available online: https://www.idc.com/getdoc.jsp?containerId=prUS44619519 (accessed on June 2, 2020)

102. B. Durakovic. Design for additive manufacturing: Benefits, trends and challenges. *Period. eng. nat. sci. Periodicals of Engineering and Natural Sciences*, 2018, Vol. 6, No. 2, pp. 179-191. DOI: 10.21533/pen.v6i2.224.

103. AMM Sharif Ullah, Tashi, Akihiko Kubo, and Khalifa H. Harib. Tutorials for Integrating 3D Printing in Engineering Curricula. 2020, Vol. 10, No. 8, pp. 194. DOI: 10.3390/educsci10080194.

104. ISO/ASTM International. Standard specification for additive manufacturing file format (AMF) Version 1.1, ISO/ASTM 52915:2020, Geneva, 2020,

105. Anton Wiberg, Johan Persson, and Johan Ölvander. Design for additive manufacturing – a review of available design methods and software. *Rapid Prototyping Journal*, 2019, Vol. 25, No. 6, pp. 1080-1094. DOI: 10.1108/RPJ-10-2018-0262.

106. 7 Families of Additive Manufacturing. Available online: https://www.additivemanufacturing.media/cdn/cms/7_families_print_version.pdf (accessed on October 20, 2020)

107. Mario Hirz, Patrick Rossbacher, and Jana Gulanová. Future trends in CAD – from the perspective of automotive industry. *Computer-Aided Design and Applications*, 2017, Vol. 14, No. 6, pp. 734-741. DOI: 10.1080/16864360.2017.1287675.

108. Yee Ling Yap, Yong Sheng Edgar Tan, Heang Kuan Joel Tan, Zhen Kai Peh, Xue Yi Low, Wai Yee Yeong, Colin Siang Hui Tan, and Augustinus Laude. 3D printed bio-models for medical applications. *Rapid Prototyping Journal*, 2017, Vol. 23, No. 2, pp. 227-235. DOI: 10.1108/RPJ-08-2015-0102.

109. Wei Sun, Binil Starly, Jae Nam, and Andrew Darling. Bio-CAD modeling and its applications in computer-aided tissue engineering. *Computer-Aided Design*, 2005, Vol. 37, No. 11, pp. 1097-1114. DOI: 10.1016/j.cad.2005.02.002.

110. Manu S. Mannoor, Ziwen Jiang, Teena James, Yong Lin Kong, Karen A. Malatesta, Winston O. Soboyejo, Naveen Verma, David H. Gracias, and Michael C. McAlpine. 3D Printed Bionic Ears. *Nano Letters*, 2013, Vol. 13, No. 6, pp. 2634-2639. DOI: 10.1021/nl4007744.

111. Daniel Steffan and Eggbeer Dominic. A CAD and AM process for maxillofacial prostheses bar-clip retention. *Rapid Prototyping Journal*, 2016, Vol. 22, No. 1, pp. 170-177. DOI: 10.1108/RPJ-03-2014-0036.

112. Yan Luximon, Roger M. Ball, and Eric H. C. Chow. A design and evaluation tool using 3D head templates. *Computer-Aided Design and Applications*, 2016, Vol. 13, No. 2, pp. 153-161. DOI: 10.1080/16864360.2015.1084188.

113. Wonhyung Jung, Seyoun Park, and Hayong Shin. Combining volumetric dental CT and optical scan data for teeth modeling. *Computer-Aided Design*, 2015, Vol. 67-68, pp. 24-37. DOI: 10.1016/j.cad.2015.04.008.

114. Ruth Jill Urbanic. From thought to thing: using the fused deposition modeling and 3D printing processes for undergraduate design projects. *Computer-Aided Design and Applications*, 2016, Vol. 13, No. 6, pp. 768-785. DOI: 10.1080/16864360.2016.1168219.

115. Letnikova Galina and Xu Na. Academic library innovation through 3D printing services. *Library Management*, 2017, Vol. 38, No. 4/5, pp. 208-218. DOI: 10.1108/LM-12-2016-0094.

116. Michele Moorefield Lang Heather. Makers in the library: case studies of 3D printers and maker spaces in library settings. *Library Hi Tech*, 2014, Vol. 32, No. 4, pp. 583-593. DOI: 10.1108/LHT-06-2014-0056.

117. Somlak Wannarumon and Erik L. J. Bohez. A New Aesthetic Evolutionary Approach for Jewelry Design. *Computer-Aided Design and Applications*, 2006, Vol. 3, No. 1-4, pp. 385-394. DOI: 10.1080/16864360.2006.10738477.

118. 3D Printed Clothing. Available online: https://3dprint.com/tag/3d-printed-clothing/ (accessed on October 20, 2020)

119. 3D printed shoes: what's available on the market today? Available online: https://www.3dnatives.com/en/3d-printed-shoes-whats-available-on-the-market-today/ (accessed on October 20, 2020)

120. 3D Printed Fashion: 10 Amazing 3D Printed Dresses. Available online: https://i.materialise.com/blog/en/3d-printed-fashion-dresses/ (accessed on October 20, 2020)

121. Nike debuts first-ever football cleat built using 3D printing technology. Available online: https://news.nike.com/news/nike-debuts-first-ever-football-cleat-built-using-3d-printing-technology (accessed on October 2020, 2020)

122. Jie Sun, Weibiao Zhou, Liangkun Yan, Dejian Huang, and Lien Ya Lin. Extrusion-based food printing for digitalized food design and nutrition control. *Journal of Food Engineering*, 2018, Vol. 220, No. Supplement C, pp. 1-11. DOI: 10.1016/j.jfoodeng.2017.02.028.

123. Clarice Lin. 3D Food Printing: A Taste of the Future. *Journal of Food Science Education*, 2015, Vol. 14, No. 3, pp. 86-87. DOI: 10.1111/1541-4329.12061.

124. Isaac Alves Ferreira and Jorge Lino Alves. Low-cost 3D food printing. *Ciência & Tecnologia dos Materiais*, 2017, Vol. 29, No. 1, pp. e265-e269. DOI: 10.1016/j.ctmat.2016.04.007.

125. Hao Jiang, Luyao Zheng, Yanhui Zou, Zhaobin Tong, Shiyao Han, and Shaojin Wang. 3D food printing: main components selection by considering rheological properties. *Critical Reviews in Food Science and Nutrition*, 2018, pp. 1-13. DOI: 10.1080/10408398.2018.1514363.

126. Deborah Lupton and Bethaney Turner. Food of the Future? Consumer Responses to the Idea of 3D-Printed Meat and Insect-Based Foods. *Food and Foodways*, 2018, No., pp. 1-21. DOI: 10.1080/07409710.2018.1531213.

127. Mehmet Sakin and Yusuf Caner Kiroglu. 3D Printing of Buildings: Construction of the Sustainable Houses of the Future by BIM. *Energy Procedia*, 2017, Vol. 134, No. Supplement C, pp. 702-711. DOI: 10.1016/j.egypro.2017.09.562.

128. Izabela Hager, Anna Golonka, and Roman Putanowicz. 3D Printing of Buildings and Building Components as the Future of Sustainable Construction? *Procedia Engineering*, 2016, Vol. 151, No. Supplement C, pp. 292-299. DOI: 10.1016/j.proeng.2016.07.357.

129. Sungwoo Lim, Richard A Buswell, Thanh T. Le, Simon A Austin, Alistair Gibb, and Antony Thorpe. Developments in construction-scale additive manufacturing

processes. *Automation in Construction*, 2012, Vol. 21, No. Supplement C, pp. 262-268. DOI: 10.1016/j.autcon.2011.06.010.

130. Yi Wei Daniel Tay, Biranchi Panda, Suvash Chandra Paul, Nisar Ahamed Noor Mohamed, Ming Jen Tan, and Kah Fai Leong. 3D printing trends in building and construction industry: a review. *Virtual and Physical Prototyping*, 2017, Vol. 12, No. 3, pp. 261-276. DOI: 10.1080/17452759.2017.1326724.

131. Conner Brett P., Manogharan Guha P., and Meyers Kerry L. An assessment of implementation of entry-level 3D printers from the perspective of small businesses. *Rapid Prototyping Journal*, 2015, Vol. 21, No. 5, pp. 582-597. DOI: 10.1108/RPJ-09-2014-0132.

132. Holzmann Patrick, Breitenecker Robert J., Soomro Aqeel A., and Schwarz Erich J. User entrepreneur business models in 3D printing. *Journal of Manufacturing Technology Management*, 2017, Vol. 28, No. 1, pp. 75-94. DOI: 10.1108/JMTM-12-2015-0115.

133. Bosqué Camille. What are you printing? Ambivalent emancipation by 3D printing. *Rapid Prototyping Journal*, 2015, Vol. 21, No. 5, pp. 572-581. DOI: 10.1108/RPJ-09-2014-0128.

134. Ikuya Fukumoto and Osamu Kondo. Three-dimensional craniofacial variation and occlusal wear severity among inhabitants of Hokkaido: comparisons of Okhotsk culture people and the Ainu. *Anthropological Science*, 2010, Vol. 118, No. 3, pp. 161-172. DOI: 10.1537/ase.091222.

135. Kenya Kura, Elijah L. Armstrong, and Donald I. Templer. Cognitive function among the Ainu people. *Intelligence*, 2014, Vol. 44, pp. 149-154. DOI: 10.1016/j.intell.2014.04.001.

136. Tadashi Nishikiya. A Positive Consideration of Ainu pattern. *Bulletin of Japanese Society for the Science of Design*, 1971, Vol. 1971, No. 13, pp. 49-65. DOI: 10.11247/jssdj.1971.49. (In Japanese)

137. Tadashi Nishikiya. Ainu pattern its origin and composition No.3. *Bulletin of Japanese Society for the Science of Design*, 1971, Vol. 1971, No. 14, pp. 74-75. DOI: 10.11247/jssdj.1971.74. (In Japanese)

138. Hisashi Sugiyama. On the Composition of Ainu Pattern. *Bulletin of Japanese Society for the Science of Design*, 1968, Vol. 1968, No. 8, pp. 60-61. DOI: 10.11247/jssdj.1968.60. (In Japanese)

139. Nobuko Tsuda, *Ainu Shishu Nyumon*. Kuruzu: Sapporo, 2011.

140. Ainu Living Culture Reproduce Manual. Available online: https://www.ff-ainu.or.jp/ (accessed on April 9, 2019)

141. Tadahiro Chiba. Problems on Making The Landscape Guideline of Facility Complex in Akan National Park. *Summaries of technical papers of the annual meeting*, 2013, No. 2013, pp. 397-398. (In Japanese)

142. Introduction of Traditional Crafts: Ainushiriki. Available online: http://www.city.sapporo.jp/shimin/pirka-kotan/jp/kogei/ainu-siriki/ (accessed on January 15, 2018)

143. SolidWorks. Available online: https://www.solidworks.com (accessed on June 5, 2020)

144. Available online: https://www.raise3d.com (accessed on June 5, 2020)

145. Hari Kishan, *Differential Calculus*. Atlantic Publishers & Distributors Pvt. Ltd.: New Delhi, 2007.

146. Yoshiki Ujiie and Yoshiyuki Matsuoka. Macro-informatics of cognition and its application for design. *Advanced Engineering Informatics*, 2009, Vol. 23, No. 2, pp. 184-190. DOI: 10.1016/j.aei.2008.10.004.

147. Toshinobu Harada. Study of Quantitative Analysis of the Characteristics of a Curve. *Forma*, 1997, Vol. 12, No. 1, pp. 55-63.

148. Kenjiro T. Miura, Junji Sone, Atsushi Yamashita, and Toru Kaneko. *Derivation of a General Formula of Aesthetic curves*, in *8th International Conference on Humans and Computer (HC2005)*. 2005: Aizu-Wakamutsu, Japan. p. 166-171.

149. Norimasa Yoshida and Takafumi Saito. Interactive aesthetic curve segments. *The Visual Computer*, 2006, Vol. 22, No. 9, pp. 896-905. DOI: 10.1007/s00371-006-0076-5.

150. Japanese pottery and porcelain. Available online: https://en.wikipedia.org/w/index.php?title=Japanese_pottery_and_porcelain&oldid=875182192 (accessed on March 15, 2020)

151. Hokkaido Ainu Motifs and Complex Patterns. Available online: https://www.kaggle.com/tashi123/hokkaidoainu-motifs (accessed on November 25, 2020)

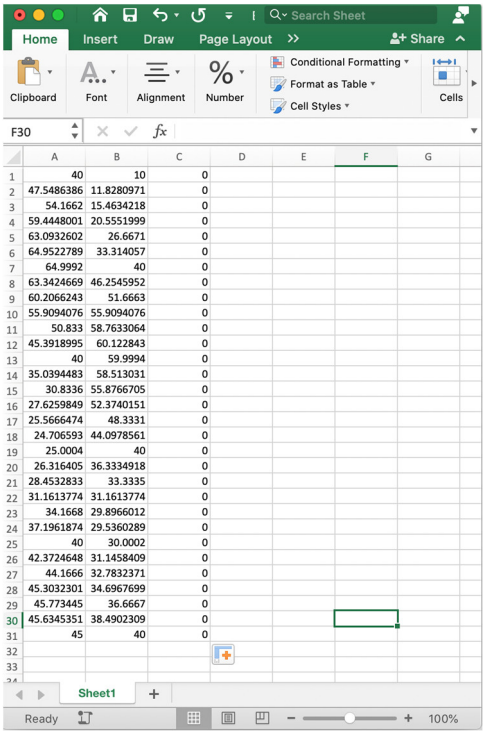
152. Hugh S Fricker. Why does charge concentrate on points? *Physics Education*, 1989, Vol. 24, No. 3, pp. 157-161. DOI: 10.1088/0031-9120/24/3/309.

Appendix A: Basic Steps of Point Cloud-based Solid CAD Modeling

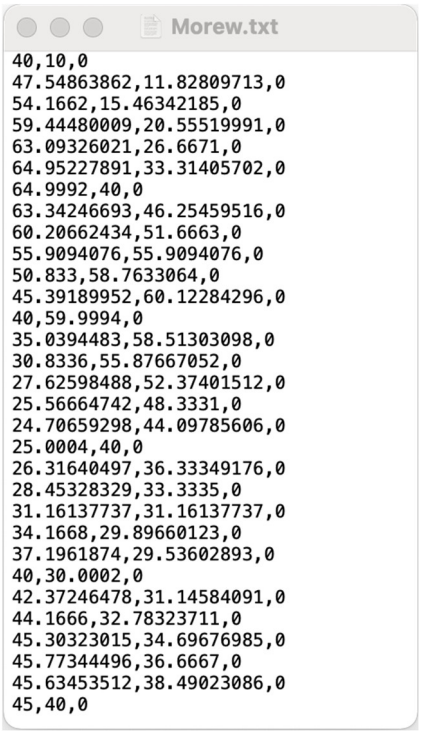
Solid CAD modeling is the modeling process to construct a solid model or solid object. In this study, a solid CAD model is created from the point cloud using SolidWorks (3D CAD software). Let consider a Morew motif of Ainu as an example for solid CAD Modeling. The steps are described as follows.

Step-1:

First, convert the point cloud dataset to a text file, as shown in Figure A.1. The text files are compatible to export the point cloud dataset in SolidWorks. When the shape is model using several point clouds, the individual text file must be created for each point cloud, and then upload the text files one-by-one. It helps give separate entities that are useful for applying any functions offered by the 3D CAD system.



(a) Point cloud dataset



(b) Text file

Figure A.1. Conversion of point cloud dataset into a text file.

Step-2:

Open the SolidWorks, and then the window shown in Figure A.2 will appear. Now select "Part" and then click on the "Ok" button, as shown in Figure A.2.

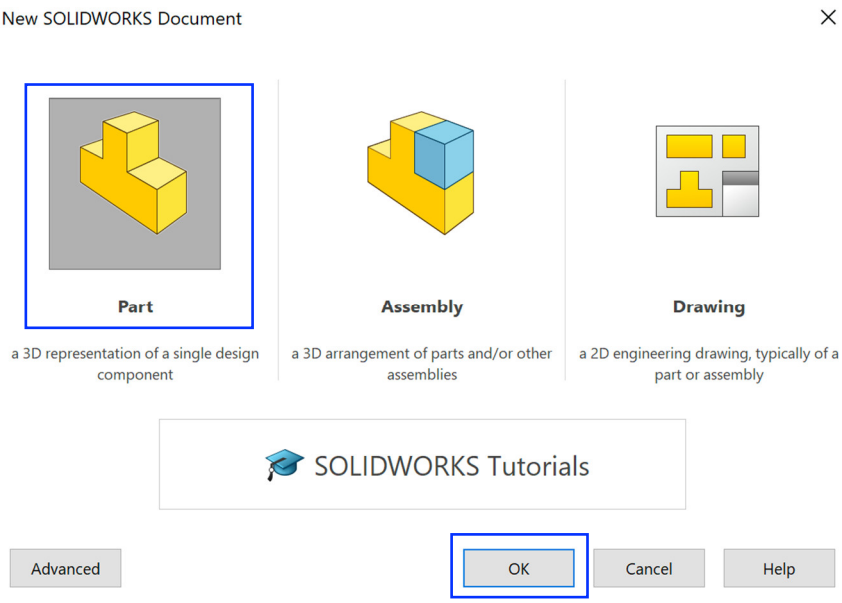


Figure A.2. Start menu of SolidWorks.

Step-3:

Click on the "Curves" menu and select the "Curve Through XYZ Points" from the drop list, as shown in Figure A.3.

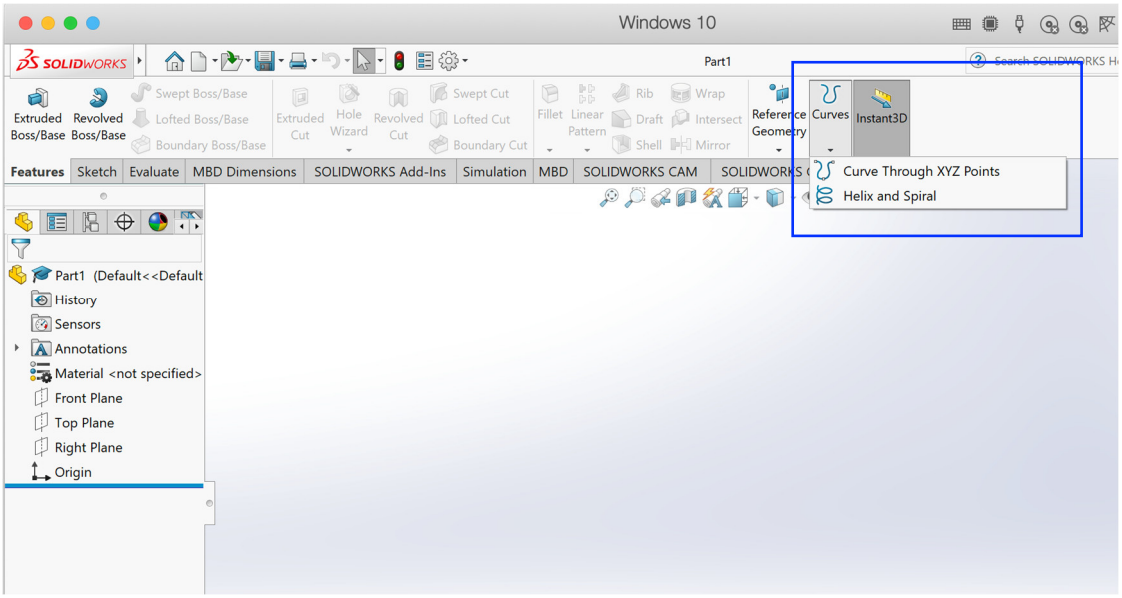


Figure A.3. Menu to import the point clouds in SolidWorks.

Step-4:

A new window will appear, as shown in Figure A.4. Now, select file type as "Text File (*.txt)," and then select the file (e.g., d1). Now, click on the "Open" button, as shown in Figure A.4.

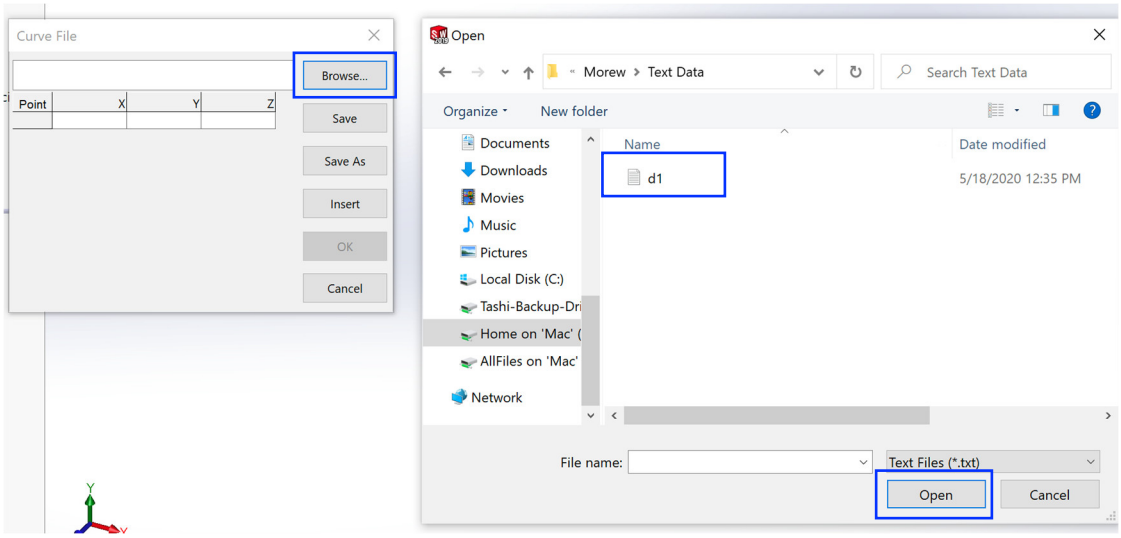


Figure A.4. Steps to open a text file in SolidWorks.

Step-5:

Click on the "Ok" button to load the text file data in SolidWorks, as shown in Figure A.5.

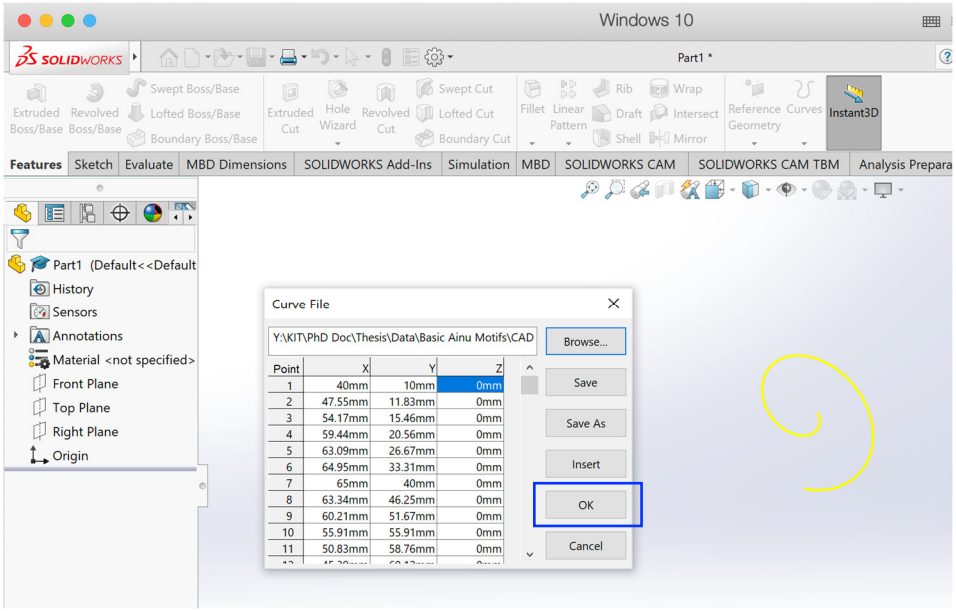


Figure A.5. Loading text data in SolidWorks.

Step-6:

Select first the plane, then click on the "Sketch" bar and click on the "Sketch" menu to sketch the shape, as shown in Figure A.6.

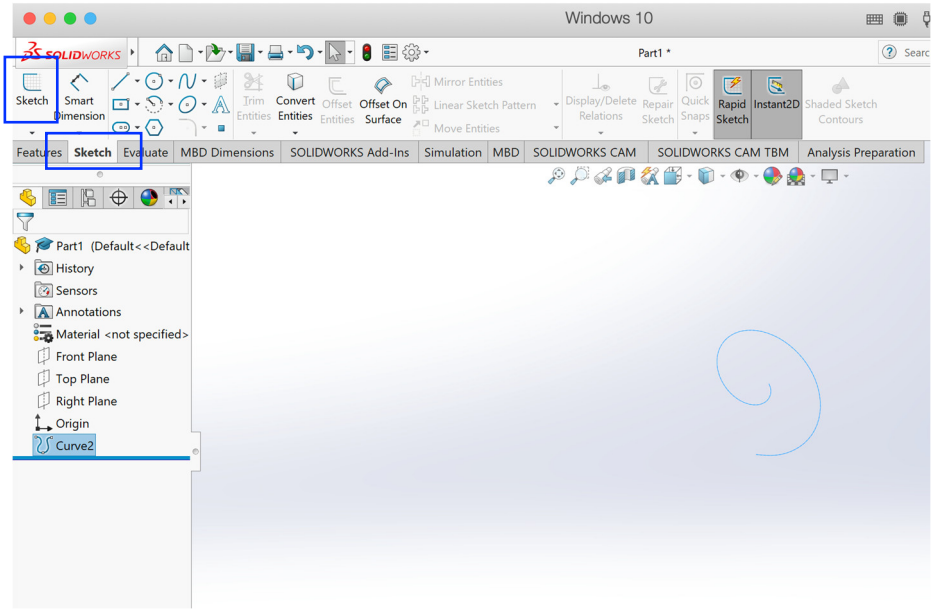


Figure A.6. Sketch preparation in SolidWorks.

Step-7:

Click on the "Convert Entities" menu and then click on the "Convert Entities" from the drop list to convert the text file data into geometric entities, as shown in Figure A.7.

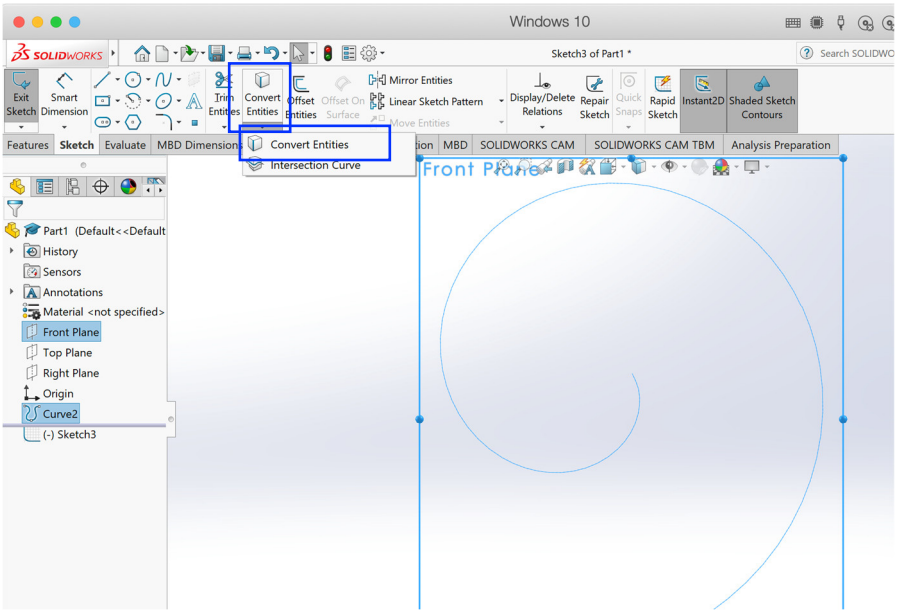


Figure A.7. Conversion of the text file data into geometric entities (i.e., curve) in SolidWorks.

Step-8:

Select the curve and then click on the "Offset menu." Fill in the offset value and select all the required parameters. Click on the checkmark to confirm the offset, as shown in Figure A.8.

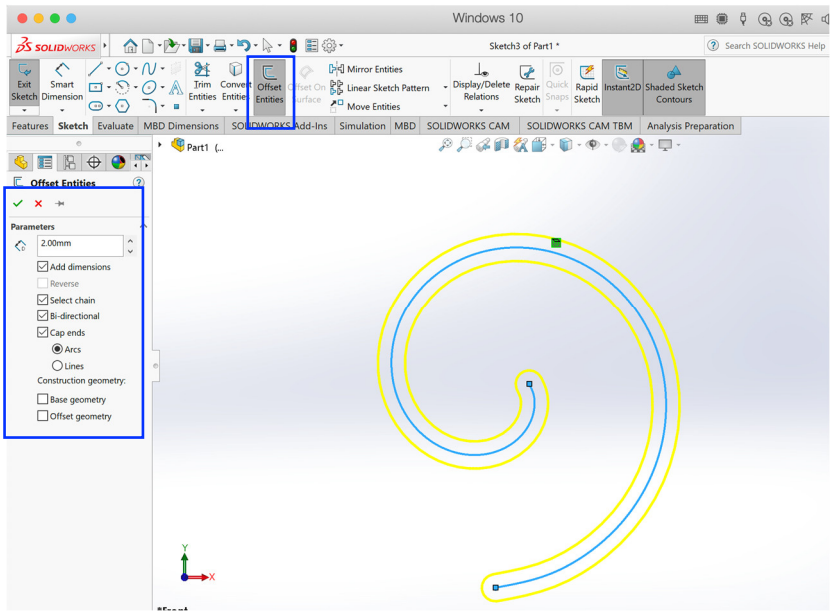


Figure A.8. Steps to create offset in SolidWorks.

Step-9:

Click on the "Feature" bar and then click on the "Extrude Boss/Bass" menu, as shown in Figure A.9.

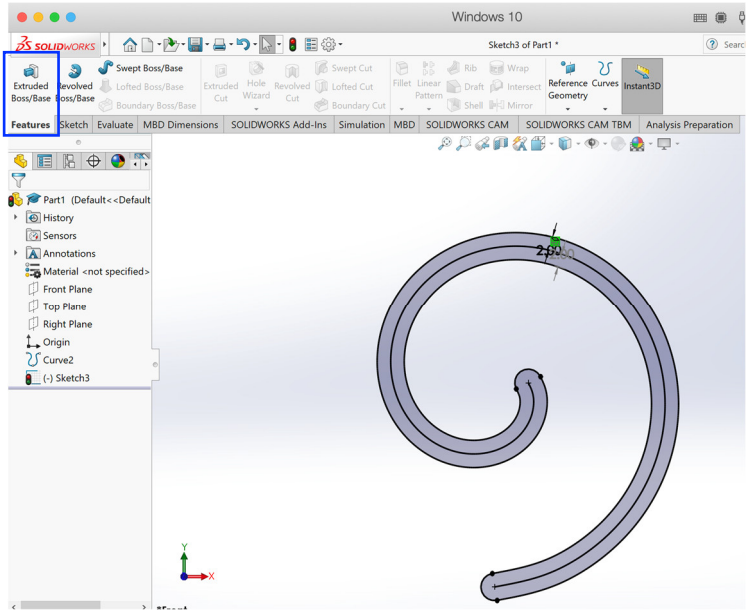


Figure A.9. Extrude preparation in SolidWorks.

Step-10:

Fill in the value of extrusion, select required parameters, and then select the regions to be extruded. Click on "✓" to confirm the extrusion function, as shown in Figure A.9.

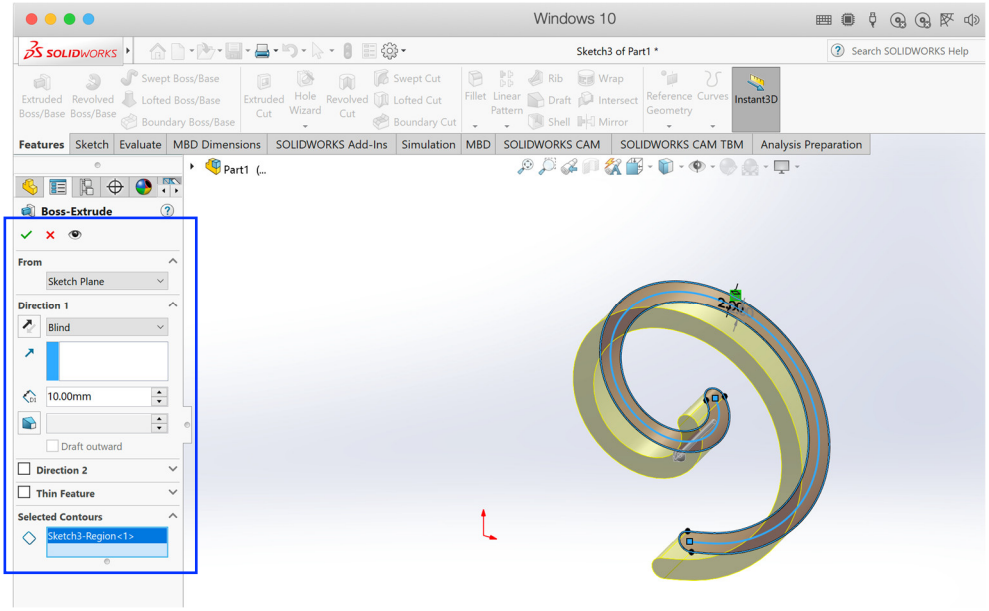


Figure A.10. Steps to extrude the surfaces in SolidWorks.

Step-11:

Figure A.11 shows the final result of solid CAD modeling. Save the solid CAD model.

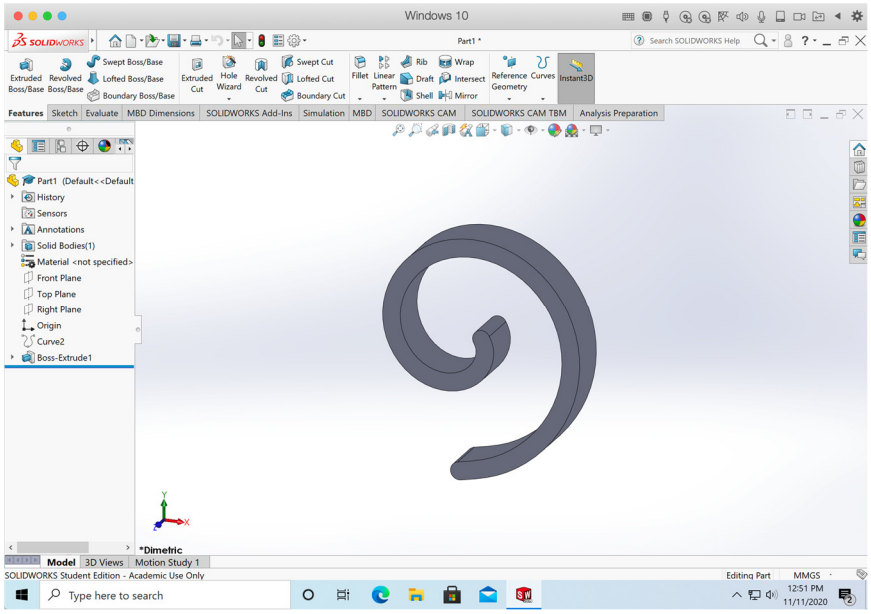


Figure A.11. A solid CAD model (virtual model).

Appendix B: Basic Steps of Triangulation

The triangulation is performed in the SolidWorks, and the basic steps are as follows.

Step-1:

In SolidWorks, click on the "Save" icon and then click on the "Save As" from the drop list, as shown in Figure B.1.

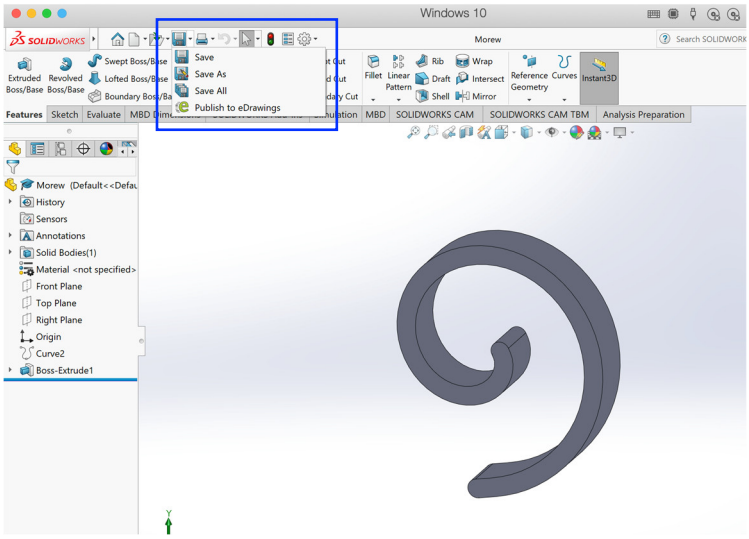


Figure B.1. Selection of a file type to be saved.

Step-2:

Given file name, then select first save as type as "STL (*.stl)" from the drop list, and click on the "Save" button, as shown in Figure B.2.

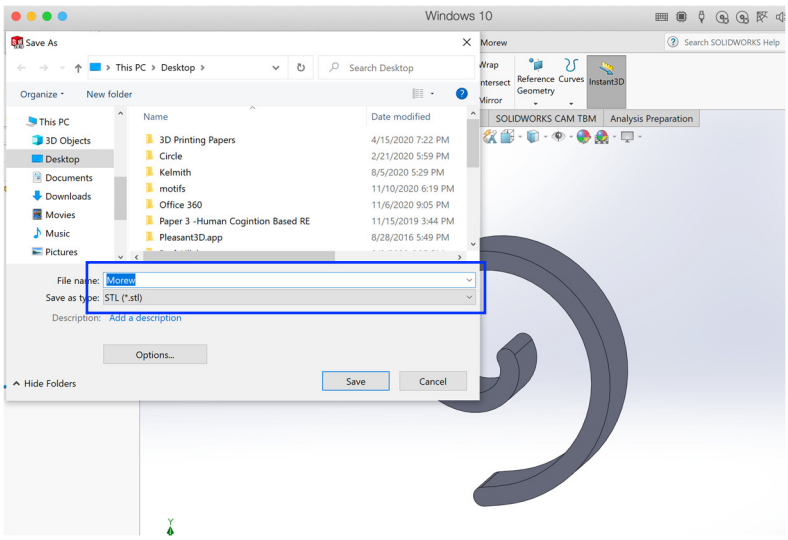


Figure B.2. Save a solid model as an STL file.

Step-3:

Click on the "Yes" button to save the solid CAD model as an STL file. Figure B.3 shows the triangulated model of the solid CAD model. Figure B.5 shows a portion of STL data of the triangulated model.

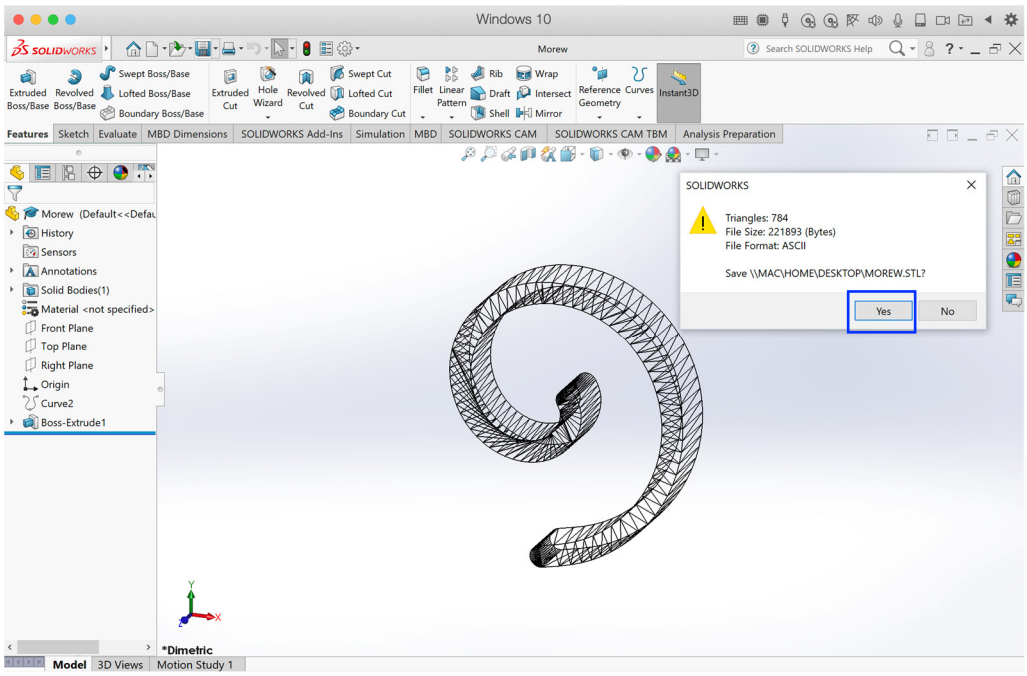


Figure B.3. A triangulated model of the solid CAD model.

STL-Data — Edited

```
solid MOREW
   facet normal -9.083476e-01 -4.182160e-01 -0.000000e+00
      outer loop
         vertex 4.410517e+01 3.955359e+01 1.000000e+01
         vertex 4.410517e+01 3.955359e+01 0.000000e+00
         vertex 4.455305e+01 3.858081e+01 0.000000e+00
      endloop
   endfacet
   facet normal -9.083476e-01 -4.182160e-01 0.000000e+00
      outer loop
         vertex 4.410517e+01 3.955359e+01 1.000000e+01
         vertex 4.455305e+01 3.858081e+01 0.000000e+00
         vertex 4.455305e+01 3.858081e+01 1.000000e+01
      endloop
   endfacet
   facet normal -9.610557e-01 -2.763549e-01 -0.000000e+00
      outer loop
         vertex 4.455305e+01 3.858081e+01 1.000000e+01
         vertex 4.455305e+01 3.858081e+01 0.000000e+00
         vertex 4.472803e+01 3.797230e+01 -1.734723e-15
      endloop
   endfacet
   facet normal -9.610557e-01 -2.763549e-01 0.000000e+00
      outer loop
         vertex 4.455305e+01 3.858081e+01 1.000000e+01
         vertex 4.472803e+01 3.797230e+01 -1.734723e-15
         vertex 4.472803e+01 3.797230e+01 1.000000e+01
      endloop
   endfacet
```

Figure B.4. A portion of STL data of a triangulated model.

Appendix C: Shape Definitions

C.1. Relevant Shapes

Let the point clouds of a circle, ellipse, and spiral be modeled by three parametric-based equations. Each shape is given by a point cloud consisting of the points $P_i = \{(P_{xi}, P_{yi}) \mid i = 0, 1, \ldots, n$. The mathematical formulations of the shapes are as follows.

Consider a circle with radius a, where $a > 0$. Let $P_c = (P_{cx}, P_{cy})$ be the center of the circle on the x–y plane. An arbitrary point on the circumference of the circle is denoted as $P_i = \{(P_{xi}, P_{yi}) \mid i = 0, 1, \ldots, n\}$ and is given by

$$P_{xi} = P_{cx} + a\cos\theta_i, \qquad P_{yi} = P_{cy} + a\sin\theta_i. \tag{C.1}$$

In Equation (C.1), $\theta_i = 2\pi i/n$. While creating a point cloud for a circle using Equation (C.1), the parameters are set as follows: $P_{cx} = 150$; $P_{cy} = 150$; $a = 100$; $n = 48$; and $i = 0, 1, ..., 48$. The point cloud shown in Figure 3-15a presenting the circle is created using the abovementioned formulation.

Consider an ellipse with major radius a and minor radius b, where $a, b > 0$ and $a > b$. Let $P_c = (P_{cx}, P_{cy})$ be the center of the ellipse in the x–y plane. An arbitrary point on the circumference of the ellipse is denoted as $P_i = \{(P_{xi}, P_{yi}) \mid i = 0, 1, \ldots, n\}$ and is given by

$$P_{xi} = P_{cx} + a\cos\theta_i, \qquad P_{yi} = P_{cy} + b\sin\theta_i. \tag{C.2}$$

In Equation (C.2), $\theta_i = 2\pi i/n$. While creating a point cloud for an ellipse using Equation (C.2), the parameters are set as follows: $P_{cx} = 75$; $P_{cy} = 75$; $a = 55$; $b = 45$; $n = 48$; and $i = 0, 1, ..., 48$. The point cloud shown in Figure 3-16(a) presenting the ellipse is created using the abovementioned formulation.

Consider a spiral. Let it be an Archimedean spiral. Let $P_C = (P_{cx}, P_{cy})$ be the center of the spiral in the x–y plane. Consider a to be the arbitrary positive real constant that controls the distance between the successive turnings of the spiral. An arbitrary point on the circumference of the spiral is denoted as $P_i = \{(P_{xi}, P_{yi}) \mid i = 0, 1, \ldots, n\}$ and is given by

$$P_{xi} = P_{cx} + a\theta_i\cos\theta_i, \qquad P_{yi} = P_{cy} + a\,\theta_i\sin\theta_i. \tag{C.3}$$

In Equation (C.3), $\theta_i = 6\pi i/n$. While creating a point cloud for a spiral using Equation (C.3), the parameters are set as follows: $P_{cx} = 600$; $P_{cy} = 600$; $a = 28.647$; $n = 48$; and $i = 0, 1, ..., 48$. The point cloud shown in Figure 3-17(*a*) presenting the spiral is created using the abovementioned formulation.

The pattern shown in Figure 4-8 is obtained by rotating five times of basic motifs. In this case, one can rotate the point cloud around the center of the shape to create other point clouds, as required. To be more specific, let $P_r = (P_{rx},P_{ry})$ be the center of the shape and $P'(t) = (P'_x(t),P'_y(t))$ be the position of the point, $P(t) = (P_x(t),P_y(t))$ after its rotation around $P_r = (P_{rx},P_{ry})$ in the anti-clockwise direction by an angle θ. Thus, the following formulation holds:

$$\begin{aligned} P'_x(t) &= P_{rx} + (P_x(t) - P_{rx})\cos\theta - \left(P_y(t) - P_{ry}\right)\sin\theta \\ P'_y(t) &= P_{ry} + (P_x(t) - P_{rx})\sin\theta + \left(P_y(t) - P_{ry}\right)\cos\theta \end{aligned} \tag{C.4}$$

Equation (C.4) can be used for creating other point clouds. For example, five other point clouds can be created by using Equation (C.4), for $\theta = \pi/3, 2\pi/3, \pi, 4\pi/3$, and $5\pi/3$. This way, one can create any shape in terms of point clouds.

C.2. Radius of Curvature

Let ρ_i be the instantaneous radius of curvature for point $P_i = \{(P_{xi}, P_{yi}) \mid i = 0, 1, \ldots, n\}$. Thus, the following relationship holds [145]:

$$\rho_i = \frac{((\dot{x}_i)^2 + (\dot{y}_i)^2)^{3/2}}{\left|\left((\dot{x}_i)(\ddot{y}_i) - (\dot{y}_i)(\ddot{x}_i)\right)\right|}. \tag{C.5}$$

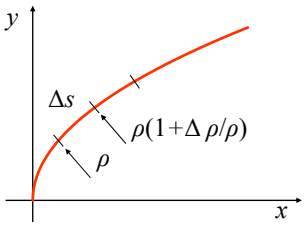
In Equation (C.5), $\dot{x}_i = P_{xi+1} - P_{xi}$; $\dot{y}_i = P_{yi+1} - P_{xy}$; $\ddot{x}_i = \dot{x}_{i+1} - \dot{x}_i$; and $\ddot{y}_i = \dot{y}_{i+1} - \dot{y}_i$. The radii of curvature for the shapes described above are computed using Equation (C.5).

C.3. Aesthetic Value

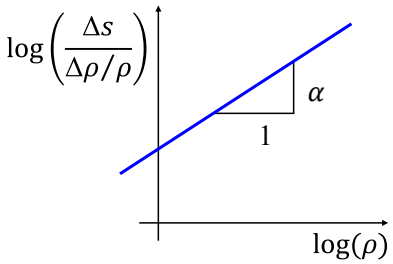
A parameter called aesthetic value, denoted as α, quantifies the aesthetic nature of a curve [147-149], which is driven from the radius of curvature (Equation C.5) and the arc length of a given curve, as shown in Figure C.1(*a*). As seen in Figure C.1(*a*), ρ, Δs, and $\Delta\rho$ are the radius of curvature, arc length, and change in the radius of curvature, respectively. The logarithmic plot of $\Delta s/ (\Delta\rho/\rho)$ versus ρ, which is called Logarithmic Curvature Histogram (LCH), becomes a straight line, and its slope is used to calculate the value of α, as seen in Figure C.1b. Therefore, the following relationship holds [148]:

$$\log\left(\frac{\Delta s}{\Delta\rho/\rho}\right) = \alpha\log(\rho) + c. \tag{C.6}$$

Where c is a constant and by knowing the value of α, the degree of aesthetics of a given curve (shape) can be quantified.



(a) Parameters for determining the aesthetic value



(b) Calculating aesthetic value using LCH

Figure C.1. Determining aesthetic value.

Acknowledgments

I would like to express my sincere gratitude to my Academic Advisor, Professor AMM Sharif Ullah at the Kitami Institute of Technology, for his patience, motivation, guidance, and continuous support. Without his excellent and professional contribution, the study would not have been successful. I could not have asked a better advisor and mentor for my doctoral study. At the same time, I am grateful to my Deputy Academic Advisors, Associate Professor Michiko Watanabe and Professor Kazuhiro Hayashida at the Kitami Institute of Technology, for their constant support and inspiration.

I would like to thank my Thesis Examination Committee Members Professor AMM Sharif Ullah (Chair), Associate Professor Michiko Watanabe, Professor Kazuhiro Hayashida, Professor Fumito Masui, Associate Professor Kazunori Takai from the Kitami Institute of Technology, and Professor Koji Teramoto from the Muroran Institute of Technology, for their encouragement and insightful comments. It encouraged me to widen my research and revise my thesis.

I would like to acknowledge the Kitami Institute of Technology for providing the materials and financial support. I am grateful to the Japan Educational Exchanges and Services (JEES) and the Japan Student Services Organization (JASSO) for granting me the scholarships.

I am thankful to Assistant Professor Akihiko Kubo and Technical Staff member Mr. Tsuyoshi Sugino at the Advanced Manufacturing Engineering Laboratory, Kitami Institute of Technology, for sharing their expertise and support. My appreciations go to my fellow lab mates for stimulating discussions and all the fun we have had in the last three years.

Before concluding, I also like to extend my sincere gratitude to the president and the management team of the College of Science and Technology. They have always been supportive and generous.

Last but not least, I would like to thank my wife, Mrs. Ugyen Lhamo, who has been my source of encouragement and pillar of support. This study is dedicated to my son Kelmith Tashi and daughter Samaya Tashi.

List of Achievements

Refereed original articles in international journals related to the thesis

1. **Tashi**, AMM Sharif Ullah, Michiko Watanabe, and Akihiko Kubo. Analytical Point-Cloud Based Geometric Modeling for Additive Manufacturing and Its Application to Cultural Heritage Preservation, Applied Sciences, Vol. 8, No. 5, pp. 656, 2018. [MDPI] [10.3390/app8050656] [SCIE Indexed] IF = 2.474 (Published in May 2018)
2. **Tashi** and AMM Sharif Ullah. Symmetrical Patterns of Ainu Heritage and Their Virtual and Physical Prototyping, Symmetry, Vol. 11, No. 8, pp. 985, 2019. [MDPI] [10.3390/sym11080985] [SCIE Indexed] IF = 2.645 (Published in August 2019)
3. **Tashi** and AMM Sharif Ullah and A. Kubo. Geometric Modeling and 3D Printing Using Recursively Generated Point Cloud, Mathematical and Computational Applications, Vol. 24, No. 3, pp. 83, 2019. [MDPI] [10.3390/mca24030083] [ESCI Indexed] IF = none (Published in September 2019)
4. AMM Sharif Ullah, **Tashi**, A. Kubo, and K. H. Harib. Tutorials for Integrating 3D Printing in Engineering Curricula, Education Sciences, Vol. 10, No. 8, pp. 194, 2020. [MDPI] [10.3390/educsci10080194] [ESCI Indexed] IF = none (Published in July 2020)

Refereed full-length articles in international conference proceedings related to the thesis

5. **Tashi**, AMM Sharif Ullah, Akihiko Kubo, and Michiko Watanabe. On the Additive Manufacturing of Hokkaido-Ainu Motifs, Proceedings of the 17th International Conference on Precision Engineering (ICPE 2018), November 12–16, 2018, Kamakura, Japan. [CD–ROM] [Paper No. B–5–4].
6. **Tashi**, AMM Sharif Ullah, Michiko Watanabe, and Akihiko Kubo. Point-cloud Based Geometric Modeling of Hokkaido-Ainu Motifs and Its Manufacturing Method, Proceedings of the 22nd Asia Pacific Symposium on Intelligent and Evolutionary Systems (IES2018), December 20–22, 2018, Sapporo, Japan. [CD–ROM] [pp. 6–13].
7. **Tashi**, AMM Sharif Ullah, and Akihiko Kubo. Developing a Human-cognition-based Reverse Engineering Approach, Proceedings of the JSME 2020 Conference on Leading Edge Manufacturing/Materials and Processing, Virtual, Online. September 3, 2020. V001T01A003. ASME. DOI: 10.1115/LEMP2020-8528