The 6th International Conference on Construction Engineering and Project Management (ICCEPM 2015)
Oct. 11 (Sun) ~ 14 (Wed) 2015 • Paradise Hotel Busan • Busan, Korea
www.iccepm2015.org

Feature Template-Based Sweeping Shape Reverse Engineering Algorithm using a 3D Point Cloud

Tae Wook Kang[1] , Ji Eun Kim[2], Chang Hee Hong[3] and Cho Gun Hwa[4]

***Abstract:** This study develops an algorithm that automatically performs reverse engineering on three-dimensional (3D) sweeping shapes using a user's pre-defined feature templates and 3D point cloud data (PCD) of sweeping shapes. Existing methods extract 3D sweeping shapes by extracting points on a PCD cross section together with the center point in order to perform curve fitting and connect the center points. However, a drawback of existing methods is the difficulty of creating a 3D sweeping shape in which the user's preferred feature center points and parameters are applied. This study extracts shape features from cross-sectional points extracted automatically from the PCD and compared with pre-defined feature templates for similarities, thereby acquiring the most similar template cross-section. Fitting the most similar template cross-section to sweeping shape modeling makes the reverse engineering process automatic.*

***Keywords:** 3D Point cloud, Reverse engineering, Rule set, Similarity, Sweeping shape*

I. INTRODUCTION

Reverse engineering is a core technology needed for the maintenance and operation of a facility or energy and for modular construction and facility renovation. The market size in relation to facility management system development in which reverse engineering is needed will reach KRW 6.8 trillion[1]. Internationally, there is an increase in the number of facility management projects using reverse engineering, such as hospital renovations [2]. In recent years, MEP (mechanical, electrical, and plumbing) facility construction and management costs have increased steadily. MEP systems have many sweeping shapes that can be modeled using cross-sections and paths. But, it is difficult to perform reverse engineering of sweeping shapes with changing cross-sections. The purpose of this study is to propose an FTS (feature template similarity)-based shape reverse engineering method. This method is applied to sweeping shape modeling to solve the mentioned problems effectively.

II. RESEARCH OBJECTIVE

To develop our FTS-based sweeping shape reverse engineering algorithm, related current research trends are studied and their characteristics are analyzed to compare and investigate the respective technologies. Based on the literature review, an algorithm is designed to solve the problems above, and a data structure required for the algorithm is developed. In order to verify the performance of the proposed method, a prototype is developed, and the algorithm performance criteria and samples required for performance evaluation are defined and evaluated.

III. RECENT RESEARCH TRENDS

There was a study on the use of features of building components to extract a building model conducted by Ning, Zhang, Wang, and Jaeger [3]. To perform reverse engineering of pipe systems, straight-line pipes were first extracted and their cross and connection relationships were considered to extract curved pipes such as L-bows [4]. However, the method was limited in that it was difficult to perform reverse engineering of pipe systems, if straight-line pipes were not recognized. Some studies related to the present one had limitations as a result of using fitting methods that utilized cross-sectional curves. In addition, when a model user has a preferred cross-sectional model, it is difficult to model sweeping shapes using that model. We propose an FTS-based sweeping shape reverse engineering algorithm to overcome that limitation.

IV. FTS-BASED SWEEPING REVERSE ENGINEERING ALGORITHM

A. *Overall process flow of the algorithm*

Sweeping is a method of 3D shape modeling using given cross-sections and paths. Cross-sections consist of various shapes of curved segments, while paths are given as curves. Many cross-sections with different dimensions can be defined in the path. When sweeping shapes are reverse engineered in PCD, a modeler selects the closest feature selection template (FST) followed by dimension fitting and sweeping along the path. This section attempts to automate this process. The overall process flow of the algorithm proposed in this study is shown in the figure 1. Once PCD are given, they undergo the segmentation process. For each PCD segment, the central path is extracted through skeletonization, and segment sections are extracted from the path. The segment sections are delivered to the FTS computation module, in which similarity is calculated, and the closest FST is selected.

The selected FST is fitted according to the segment. The fitted FST performs sweeping on the basis of the path and creates a model automatically. The following is pseudo code of the above overall FTS algorithm.

[1] Senior researcher, Kyunggi-do, laputa99999@gmail.com (*Corresponding Author)
[2] Researcher, Kyunggi-do, kje@kict.re.kr
[3] Senior researcher, Kyunggi-do, chhong@kict.re.kr
[4] Researcher, Kyunggi-do, cgh@kict.re.kr

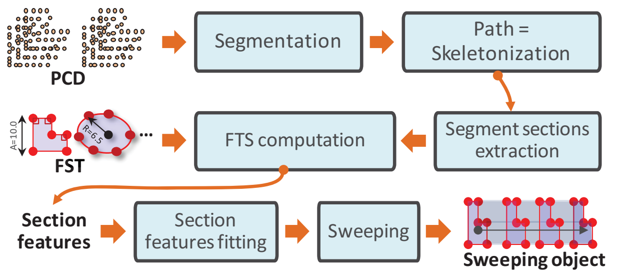


FIGURE 1. FTS-based sweeping reverse engineering algorithm workflow

Segments = Segmentation(PCD)

Paths[Segment] = Skeletonization(Segments)

Sections[Segment, Section] = SectionExtraction(Segments, Paths)

FST[Segment, Section] = FTS(FST, Sections, Paths)

FSTfitting[Segment, Section] = Fitting(FST, Sections, Paths)

B. Segmentation

Segmentation is acquired using surface smoothness constraint-based region growth [5]. The normal vector np of each point p = (px, py, pz) in PCD is calculated for segmentation according to surface smoothness similarity. To calculate it, k-NN and fixed distance neighbors (FDN) are used for a search of neighboring points of every p. The smoothness threshold is judged using || np • ns || > cos(θth).

C. Skeletonization

To extract skeletons of acquired segments, an L-medial condition-based skeleton extraction method is applied [6]. This method is more robust to PCD noise and extracts more accurate skeletons than other methods [7]. L1-medial defines the argmin$_x$ function (1) to find the value of x that has a minimum distance to given PCD $Q = \{q_j\}_{j \in J} \subset \mathbb{R}^3$.

$$\mathrm{argmin}_X \sum_{i \in I} \sum_{j \in J} \|x_i - q_j\| \theta(\|x_i - q_j\|) + R(X) \quad (1)$$

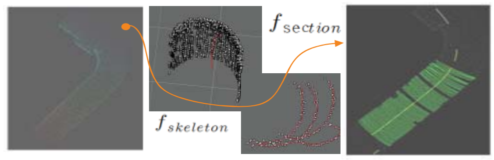
Here, I and J are the point set x index and Q set index, respectively. $\theta(r) = e^{-r^2/(\frac{h}{2})^2}$ is a corrosion function, whose corrosive value depends on the difference between the x and Q sets. R(X) is a function that calculates a repulsion force that causes no more contraction of the contracted set x defined by a corrosion function after the set x was contracted in the local center direction. The R function adjusts a repulsion force by comparing allowable contraction value and distance between local neighboring points of x. Multiple x points contracted in the center axis are combined into a single branch, and the distance between branches and the direction of branch end edge are used to merge adjacent branches. The branch acquired in the described manner is used as a sweeping path.

D. Section extraction

Sections are extracted by each branch of the path acquired in the segment. A branch is the start and end points of a curved line in a path, while a curved line is divided into an interval I_{curve} to extract a section. To extract a section, points P_{sec} are extracted as much as a section's thickness T_{sec} at a branch P_E of the path to be extracted. The $S_{plane.affine}$ transformation matrix is obtained to project 3D into 2D points in a plane S_{plane} perpendicular to the path, and P_{sec} is projected onto the plane to acquire 2D point set P_{proj}.

V. PROTOTYPE DEVELOPMENT AND TEST

FTS definition interpretation and algorithm execution modules were implemented and tested using the above-mentioned FTS definition and algorithm.



Section	X	Y	Z	R
#1	-6.481	31.983	26.940	0.646
…	…	…	…	…
#59	-5.568	31.888	25.806	0.611
#60	-6.061	31.941	25.004	0.615
Std				0.025

FIGURE 2. Prototype test result

VI. CONCLUSION

An FTS-based sweeping shape reverse engineering algorithm was proposed and its performance was evaluated. FTS-based sweeping reverse engineering algorithm can effectively solve these problems and enables reverse engineering automation. In the future, we will increase the number of cross-sectional segment types that can be processed in the FTS and verify whether they can be used in other fields of application.

ACKNOWLEDGE

This research was supported by a grant from a Research Project (Architectural MEP Object Reverse Engineering Technology Development for the Facility Management, 2015) funded by the Korea Institute of Construction Technology.

REFERENCES

[1] Ministry of Knowledge Economy, "Facility Management System Development", 2008
[2] Corble KP, "Feature: Generating BIM under pressure. Professional Survey Magazine", 2012
[3] Ning X, Zhang X, Wang Y, Jaeger M, "Segmentation of architecture shape information from 3D point cloud" In: Proceedings of the 8th International Conference on Virtual Reality Continuum and its Applications in Industry. ACM pp.127-132, 2009
[4] Kawashima K, Kanai S, Date H, "Automatic recognition of piping system from laser scanned point clouds using normal-based region growing", ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. 1, No. 2, pp.121-126, 2013
[5] Rabbani T, van den Heuvel F, "Automatic point cloud registration using constrained search for corresponding objects", In: 7th Conference on Optical, pp.3-5, 2005
[6] Huang H, Wu S, Cohen-Or D, Gong M, Zhang H, Li G, Chen B, "L1-medial skeleton of point cloud". ACM Trans Graph, Vol. 32, No.4, pp.65, 2013
[7] Cao J, Tagliasacchi A, Olson M, Zhang H, Su Z, "Point cloud skeletons via Laplacian-based contraction", Shape Modeling International Conference (SMI), pp.187-197, 2010