Title:

Shape Descriptor-based Similar Feature Extraction for Finite Element Meshing

Authors:
Hideyoshi Takashima, hideyoshi_takashima@ais-hokkaido.co.jp, Hokkaido University, Japan
Satoshi Kanai, kanai@ssi.ist.hokudai.ac.jp, Hokkaido University, Japan

Keywords:
Feature Extraction, Shape Descriptor, Key Point Matching, SHOT, Shape Index, Projective Transformation, Triangular Mesh, Finite Element Method.

DOI: 10.14733/cadconfP.2020.297-301

Introduction:

Large-scale finite element models (FE models) used in car crash simulations, etc. require high-quality FE meshes in compliance with meshing specifications, such as the location and resolution of the meshing pattern's features such as ribs and bosses to ensure simulation accuracy. However, the automatic feature-compliant meshing is not fully supported in commercial CAE software, requiring many manual operations and resulting in a high person-hour ratio to the whole CAE process. Extracting feature shapes, such as ribs and bosses, for which the meshing specifications have been strictly assigned from the complex shape of the product CAD model is particularly time consuming. Furthermore, the feature geometries are not uniquely shaped and are usually bounded by smooth and indistinct boundaries, often leading to oversights. Therefore, an automated feature extraction technique targeting a feature-compliant finite element meshing is strongly required where the features whose geometries are not identical to a reference feature shape but are similar to it can be extracted from a product's CAD model.

Researchers have proposed methods to extract feature shapes from CAD models to generate meshes of FE models [2-3][5]. However, three main problems persist. First, the feature extraction does not work robustly when the CAD model includes PDQ issues such as cracked or degenerated geometries. Second, features surrounded by complex and smooth boundaries, commonly found in casted or molded parts, remain difficult to detect. Finally, the extraction algorithm must be designed in an ad-hoc way for different feature types and even for features with similar shapes. Thus, it is difficult to apply these methods of feature extraction when developing a feature-compliant finite element meshing.

This work therefore proposes a feature extraction methodology that allows the extraction of features containing nonidentical geometries similar to a reference feature shape from a target shape. In this methodology, the reference feature shape and target shape are represented by a set of shape descriptors defined on triangular meshes; extraction is performed by finding similarities between the descriptors on the reference feature shape and those of the target shape under the projective transformation.

Similar Feature Extraction Method:

Basic Concept

The proposed similar feature extraction method exhaustively finds the features similar in shape and size to the reference feature shape pre-specified by the user from a target product shape represented by a solid model. The targeted features include ribs and bosses in casted or molded parts but are not limited to them. This methodology aims to solve the three main identified issues via the approaches described below.

- The representations of the target and reference feature shapes are converted from solid models to triangular meshes, which are used for extracting similar features. This enables a stable feature extraction even when a solid model contains PDQ-degraded geometries and/or when the feature shape boundaries are ambiguous.
- Shape descriptors defined on triangular meshes are applied to feature extraction. This approach has been demonstrated to aid in object recognition and similar shape retrieval in meshes and point clouds. Using this descriptor-based approach, the extraction algorithm can be unified even for different feature types or features with similar shapes.
- A feature to be searched is included to allow similar features to be matched with the reference shape under the projective transformation to allow for the extraction of feature shapes containing a parametric deformation relationship with the reference shape. Thus, this should allow for the extraction of features that are either identical or similar in shape to the reference shape.

Outline of Similar Feature Shape Extraction Procedure

The proposed similar feature extraction methodology comprises six steps and is summarized in Fig. 1. Steps 1 to 3 create the shape descriptors of the target and reference feature shapes; steps 4 to 6 then search for feature shapes similar to the reference shape.

Step 1: Generating triangular meshes from a solid model

A dense triangular mesh $M_D^T = \langle V_D^T, T_D^T \rangle$ (where V_D^T and T_D^T are a vertex and triangle set, respectively) and a sparse triangular mesh $M_S^T = \langle V_S^T, T_S^T \rangle$ are generated from a solid model S^T of the target shape using a CAE preprocessor. Similarly, triangular meshes $M_D^R = \langle V_D^R, T_D^R \rangle$ and $M_S^R = \langle V_S^R, T_S^R \rangle$ of the reference feature shape to be extracted are also generated.

Dense triangular meshes are necessary to calculate a feature shape descriptor with high accuracy. On the other hand, sparse triangular meshes are necessary to select a small number of distinctive feature key points with low calculation cost.

Step 2: Generating the shape descriptors

The shape index (SI) descriptor $f_{SI,i}^T$ [1] and SHOT descriptor $f_{SHOT,i}^T$ [7] are calculated at each vertex $v_{S,i}^T$ ($\in V_S^T$) of the sparse triangular mesh M_S^T to create an SI descriptor set $F_{SI}^T = \{f_{SI,i}^T\}$ and SHOT descriptor set $F_{SHOT}^T = \{f_{SHOT,i}^T\}$, respectively. To reduce the scale dependency of the SHOT descriptor, each descriptor $f_{SHOT,i}^T$ is evaluated at $v_{S,i}^T$ using multiple radius of the support sphere. Similarly, the descriptor sets $F_{SI}^R = \{f_{SI,i}^R\}$ and $F_{SHOT}^R = \{f_{SHOT,i}^R\}$ of the reference feature shape to be extracted are created.

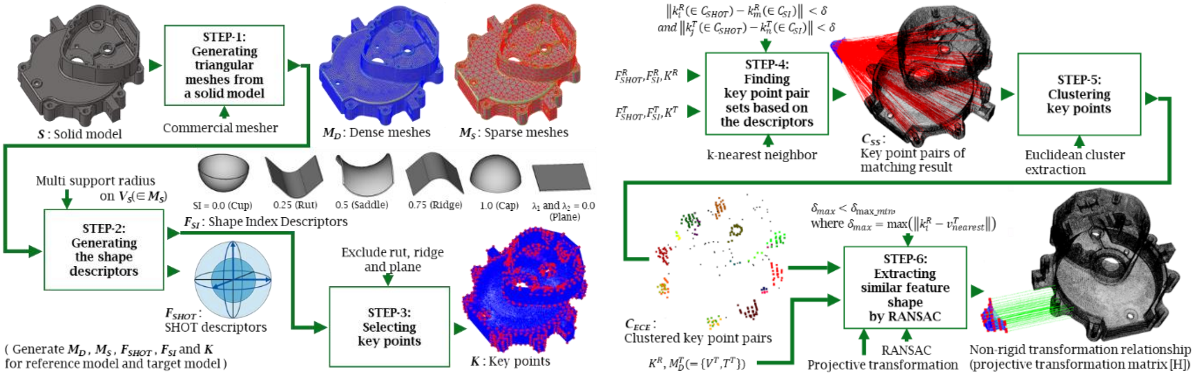
STEP-1: Generating triangular meshes from a solid model
S : Solid model
Commercial mesher
M_D : Dense meshes
M_S : Sparse meshes
Multi support radius on $V_S (\in M_S)$
STEP-2: Generating the shape descriptors
SI = 0.0 (Cup)
0.25 (Rut)
0.5 (Saddle)
0.75 (Ridge)
1.0 (Cap)
λ_1 and λ_2 = 0.0 (Plane)
F_{SI} : Shape Index Descriptors
F_{SHOT} : SHOT descriptors
Exclude rut, ridge and plane
STEP-3: Selecting key points
K : Key points
(Generate M_D, M_S, F_{SHOT}, F_{SI} and K for reference model and target model)
$\|k_i^R(\in C_{SHOT}) - k_m^R(\in C_{SI})\| < \delta$ and $\|k_j^T(\in C_{SHOT}) - k_n^T(\in C_{SI})\| < \delta$
$F_{SHOT}^R, F_{SI}^R, K^R$
$F_{SHOT}^T, F_{SI}^T, K^T$
STEP-4: Finding key point pair sets based on the descriptors
k-nearest neighbor
C_{SS} : Key point pairs of matching result
STEP-5: Clustering key points
Euclidean cluster extraction
$\delta_{max} < \delta_{max\_min}$, where $\delta_{max} = \max(\|k_i^R - v_{nearest}^T\|)$
STEP-6: Extracting similar feature shape by RANSAC
C_{ECE} : Clustered key point pairs
$K^R, M_D^T (= \{V^T, T^T\})$
RANSAC
Projective transformation
Non-rigid transformation relationship (projective transformation matrix [H])

Fig. 1: Procedure of the proposed similar feature shape extraction.

This SI descriptor $f_{SI,i}$ expresses the curvature of the local surface around a vertex v_i as a single scalar value and has scale and rotation-invariant property. The SI descriptor $f_{SI,i}$ [1] is defined from the maximum and minimum principal curvatures λ_1 and λ_2 at v_i as

$$f_{SI,i} = \frac{1}{2} - \frac{1}{\pi} tan^{-1}\left(\frac{\lambda_1 + \lambda_2}{\lambda_1 - \lambda_2}\right). \tag{1}$$

The SHOT descriptor is a 352-dimensional vector encoding the statistical distribution in the normal direction at local vertices around v_i and is rotationally invariant [7].

Step 3: Selecting key points

Key points are a subset of vertices on a triangular mesh where descriptor values for feature extraction are evaluated. Adoption of a small number of distinct feature key points has demonstrated to have increased object recognition and localization reliability and to have decreased processing time [4]. As such, when SI descriptor $f_{SI,i}^T$ at vertex $v_{S,i}^T (\in V_S^T)$ of the target shape exhibits a surface with low distinct feature; i.e., plane, rut and ridge; that vertex is not selected as key point. Only remaining vertices $v_{S,i}^T$ are adopted as key point set $K^T = \{k_i^T\}(\subset V_S^T)$. The SI and SHOT descriptors at key point $k_i^T (\in K^T)$ are adopted as feature descriptor sets $F_{SI(k)}^T = \{f_{SI(k),i}^T\}(\subset F_{SI}^T)$ and $F_{SHOT(k)}^T = \{f_{SHOT(k),i}^T\}(\subset F_{SHOT}^T)$, respectively. Similarly, key point set $K^R = \{k_i^R\}(\subset V_S^R)$ and feature descriptor sets $F_{SI(k)}^R = \{f_{SI(k),i}^R\}(\subset F_{SI}^R)$ and $F_{SHOT(k)}^R = \{f_{SHOT(k),i}^R\}(\subset F_{SHOT}^R)$ are generated for the reference feature shape.

Step 4: Finding key point pair sets based on the descriptors

For each SHOT descriptor $f_{SHOT(k),i}^T\left(\in F_{SHOT(k)}^T\right)$ at the key point k_i^T $(\in K^T)$ on the target shape, N corresponding key points are searched from the key point set $K^R = \{k_j^R\}$ on the reference feature shape according to the ascending order of distance $\|f_{SHOT(k),j}^R - f_{SHOT(k),i}^T\|$, thus creating a nearest key point pair set with respect to SHOT $C_{SHOT} = \left\{\left(k_j^R, k_i^T\right)_p \mid p \in [1, N], k_i^T \in K^T\right\}$. Similarly, the nearest key point pair set with respect to SI $C_{SI} = \{(k_n^R, k_m^T)_q \mid q \in [1, M], k_m^T \in K^T\}$ is generated by evaluating the distance $\|f_{SI(k),n}^R - f_{SI(k),m}^T\|$.

As the SI and SHOT descriptors express the local curvatures in different forms, the key point pairs that have higher similarity are then selected from C_{SHOT}. This is done by selecting key point pairs from C_{SHOT} with a high correspondence between the SHOT and SI descriptors, meaning that there is at least one nearest key point pair in C_{SI} close to a given nearest key point pair C_{SHOT} within a distance threshold δ, i.e. ($\mathrm{dist}\left(k_i^T, k_m^T\right) < \delta \wedge \mathrm{dist}\left(k_j^R, k_n^R\right) < \delta, \left(k_j^R, k_i^T\right) \in C_{SHOT}, \exists(k_n^R, k_m^T) \in C_{SI}$). If a key point pair $\left(k_j^R, k_i^T\right)$ in C_{SHOT} satisfies this condition, it is stored in the new nearest key point pair set $C_{SS} = \left\{\left(k_j^R, k_i^T\right)\right\}(\subset C_{SHOT})$.

Step 5: Clustering key points

The key points $k_i^T \in (K_{SS}^T = \left\{k_i^T \mid \left(k_j^R, k_i^T\right) \in C_{SS}\right\})$ of the target shape included in the key point pair set C_{SS} may be distributed to the multiple regions on the target shape that are similar to the reference feature shape. Thus, to increase the search efficiency of the random sample consensus (RANSAC) algorithm employed in step 6, Euclidean clustering is applied to the key points in K_{SS}^T. Additionally, the close feature key point pair set $C_{ECE} = \cup\, C_{ECE,l}$ is generated to aggregate the key points close to each other into one cluster $C_{ECE,l}$.

Step 6: Extracting similar feature shapes using a random sample consensus (RANSAC) algorithm

First, five of the key point pairs $\left\{\left(k_i^{RC}, k_j^{TC}\right) \mid k_i^{RC} \in K^{RC}, k_j^{TC} \in K^{TC}\right\}$ from one key point cluster $C_{ECE,l}$ are randomly selected. The 4×4 projective transformation matrix $[H]$, which transforms the selected key point pairs $k_i^{RC} (\in K^{RC})$ into $k_j^{TC} (\in K^{TC})$, is then estimated using Eqn. (2) and (3) via lower–upper (LU) decomposition.

$$\frac{1}{w}[H]\{Q^R\} - \{Q^T\} = \{0\}, \tag{2}$$

$$[A]^T\{H\} = \{B\}, \tag{3}$$

where $[H] = \left[h_{ij}\right]$ is a 4×4 matrix with $h_{44} = 1$, $\{Q^R\}$ and $\{Q^T\}$ are the 4×1 homogeneous coordinates of the key points k_i^{RC} and k_j^{TC}, w is the fourth component of $[H]\{Q^R\}$, i.e. $\sum_{i=1}^{4} h_{4,i} q_i^R$, $\{H\}$ is the 15×1 column vector in which all components of the matrix $[H]$ are arranged in a column vector, and $[A]$ and $\{B\}$ are the 15 × 15 matrix and 15×1 column vector calculated from the coordinates of k_i^{RC} and k_j^{TC}, respectively. Next, the transformed key point set $\widetilde{K^R}$ is generated by applying the transformation $[H]$ to all key points other than the selected five in K^{RC}. The distance between each transformed key point in $\widetilde{K^R}$ and the closest vertex of M_D^T is then evaluated, so that the estimated transformation $[H]$ represents an appropriate one. For this judgement, we first search the nearest point of $\widetilde{k_i^R}(\in \widetilde{K^R})$ from V_D^T using the k-Nearest Neighbor (kNN) algorithm and calculate its error $\|\widetilde{k_i^R} - v_{D,nearest}^T\|$. Next, if the maximum error $\delta_{max} = \max\left(\|\widetilde{k_i^R} - v_{nearest}^T\|\right)_{\widetilde{k_i^R} \in \widetilde{K^R}}$ of $\widetilde{K^R}$ is less than the threshold $\delta_{\max\_min}$, we update $\delta_{\max\_min}$ and $[H]_{min}$. If $\delta_{\max\_min}$ is not updated after the specified iterations of the updates, the estimation of $[H]_{min}$ concludes. Finally, $[H]_{min}$ gives the best projective transformation matrix that transforms the reference feature shape to the similar featured shape portion on the target shape.

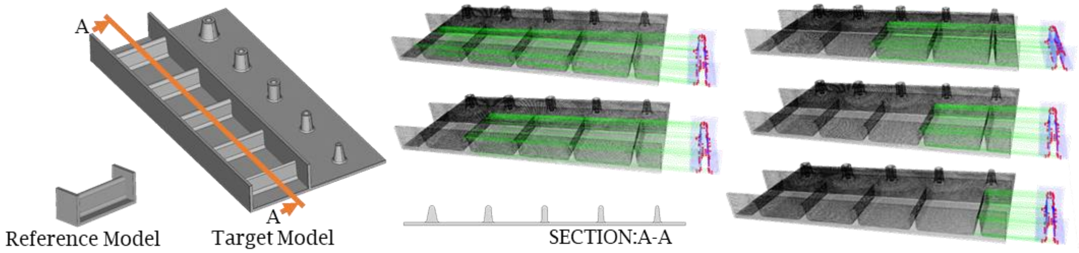

Fig. 2: Extraction results of ribs with simple shapes.

Verification of the similar feature extraction:

Extraction of similar ribs with simple shapes

To verify the proposed methodology, feature extraction was performed from a target flat-plate model containing five similar ribs with simple shapes, as shown in Fig. 2. Of the five ribs, one was identical to the reference rib, whereas the others were similar but had a varied top and bottom width. As the estimated projective transformation allowed a high degree of freedom in deformation under the originally set parameters, sometimes only a portion of the rib along its longitudinal direction was matched with the reference feature. To avoid this partial matching problem, the SI threshold was adjusted to allow the key points around the edge portions exhibiting strong geometric features to remain. Additionally, upper and lower bounds of the volume expansion ratio, scale value, and oblique distortion of the projective transformation $[H]$ were defined. As a result, the five ribs were correctly extracted, as shown in Fig. 2.

Extraction of complex features with anisotropic scale

A second verification procedure was then performed to determine if a feature shape with anisotropic scale differences from the reference feature could be extracted. A target shape close to a casted product with various feature shapes shown in Fig. 3 was selected. A portion of the target shape was cut out and deformed with anisotropic scaling to create the reference feature shape. The similar feature shape was successfully extracted, as shown in Fig. 3; however, the feasible scale range differed in the directions. Features up to 1.5 times the width and 1.2 times the length were extracted. This limitation is specified by the bounds of the distortion constraint on the projective transformation. Thus, expanding the anisotropic scale range would require extending the bounds of the constraints on the projective transformation.

Extraction of multiple similar features

A third extraction was then performed to verify the extraction of multiple features with similar shapes on the same target shape, as shown in Fig. 4. The reference feature shape was created by cutting out a local area in the target shape. Five of the six feature shapes were successfully extracted; the sixth could also be found by changing the mesh and the position of the reference feature. Future work should thus aim to improve the robustness of the extraction, possibly by increasing the mesh density, equalizing the sampling density of key points bet ween the reference feature and the target shape, and finding the most suitable position of the reference feature shape for extraction.

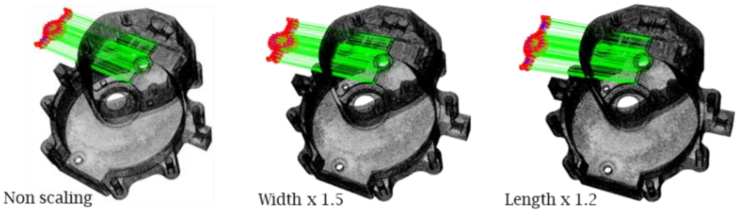

Fig. 3: Extraction results of complex features with anisotropic scales (Red: Reference, Black: Target).

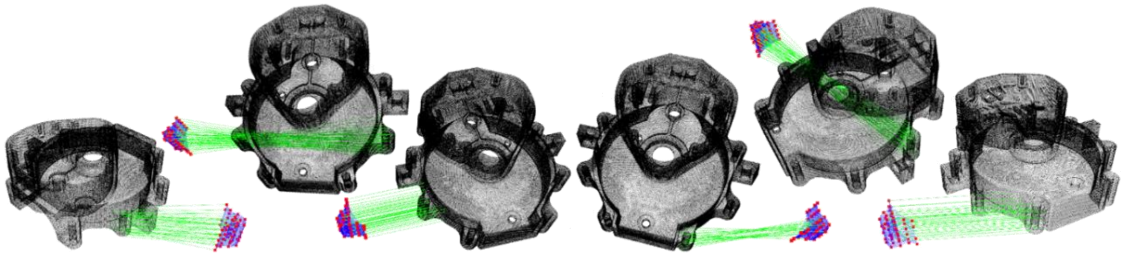

Fig. 4: Extraction results of multiple similar features.

Conclusions:

In this work, a method for extracting feature shapes similar to a reference feature from a target shape was proposed and verified for feature-compliant FE meshing. The method was based on the shape descriptor representation defined on triangular meshes, the matching operation between a reference feature shape and a target shape with the help of descriptors, and the estimation of the projective transformation between them. Future work will focus on ex tending this methodology to find the typical boss and rib shapes proposed by [6] and on developing a method that assigns an appropriate FE meshing operation to the extracted features that conforms to the specifications.

References:

[1] Itskovich, A.; Tal, A.: Surface partial matching and application to archaeology, Computers & Graphics, 35(2), 2010, 334-341. https://doi.org/10.1016/j.cag.2010.11.010

[2] Lai, J.Y.; Wang, M.H.; Song, P.P.; Hsu, C.H.; Tsai, Y.C.: Recognition and decomposition of rib features in thin-shell plastic parts for finite element analysis, Computer-Aided Design and Applications, 15(2), 2018, 264-279. https://doi.org/10.10 80/16864360.2017.1375678

[3] Lu, Y.; Gadh, R.; Tautges, T.J.: Feature based hex meshing methodology: feature recognition and volume decomposition, Computer-Aided Design, 33(3), 2001, 221-232. https://doi.org/10.1016/S0010-4485(00)00122-6

[4] Nagase, M.; Akizuki, S.; Hashimoto, M.: Reliable object recognition using 3-D feature points for minimizing its mismatching, Journal of The Japan Society for Precision Engineering, 79(11), 2013, 1058-1062. https://doi.org/10.2493/jjspe.79.1058

[5] Onodera, M.; Hariya, M.; Kongo, C.; Shintani, M.; Ka, K.; Watanuki, K.: Development of high precise similar sub-part search technique for automatic mesh generation reusing proven models, Transactions of The Japan Society of Mechanical Engineers, 85(880), 2019, 19-00138. https://doi.org/10.1299/transjsme.19-00138

[6] Takaishi, I.; Kanai, S.; Date, H.; Takashima, H.: Free-Form feature classification for finite element meshing based on shape descriptors and machine learning, Proceedings of CAD'19, 2019, 414-419. https://doi.org/10.14733/cadconfP.2019.414-419

[7] Tombari, F.; Salti, S.; Stefano, L.D.: Unique Signatures of Histograms for Local Surface Description, Proceedings of the European Conference on Computer Vision 2010 (ECCV 2010), Part 3, 2010, 356-369. https://doi.org/10.1007/978-3-642-15558-1_26