

グラフニューラルネットワークを用いた境界表現 CAD モデルからの形状特徴認識の研究

北海道大学 ○守屋遼輔, 金井理, 伊達宏昭, (株)AIS 北海道 泰地哲史, 高嶋英敏

要旨

境界表現 CAD モデル上の形状特徴の認識は, CAD/CAM/CAE 連携に必要な重要な処理であるが, 汎用性と実用性の高い形状特徴自動認識手法は未だ実現されていない. 本研究では, 境界表現 CAD モデルの位相構造がグラフ構造で表現できることを利用し, グラフ上での深層学習が直接行えるグラフニューラルネットワーク(GNN)を用いて, CAD モデル上の形状特徴を自動認識する新たな手法を開発した. 本手法は, 特にモデルの回転や平行移動に不変な認識を可能とするグラフ構成法と GNN での学習法を提案する. また, CAE 向け形状特徴であるボスやリブが含まれる実用的データセットを用いて認識性能検証を行った.

1. はじめに

設計対象の有限要素解析 (FEA) には, 対象の CAD モデルを有限要素メッシュに分割する必要がある. メッシュ分割の品質は, 解析精度に影響を与えるため, 現在, 各企業では, 解析精度保証のため, CAD モデルのメッシュ分割方法を細かく規定している. この規定中には, 複雑な形を持つボスやリブ等の形状特徴部に対するメッシュ分割仕様も数多く含まれる. このような仕様に適合するメッシュ分割を高精度・高効率に行うには, CAD モデル上の形状特徴の自動認識処理が必須となる.

近年, 認識処理の汎化能力の高さから, 境界表現 CAD モデルから深層学習により形状特徴認識を行う手法が提案されている. これらでは, 境界表現モデルをボクセルや点群・メッシュの表現に変換した上で特徴認識を行っている[1]. しかしモデル解像度の低下に伴う認識性能低下や, 学習時間の増大等の課題があった. 一方, 最近, 境界表現モデルのデータ構造と親和性が高いグラフ上での深層学習が直接可能な Graph Neural Network (GNN) による形状特徴認識手法も提案され始めたが[2,3], 自由曲面形状特徴に対する認識能力が未検証であり, また回転不変な認識が行えないといった欠点を持つ.

そこで本研究では, GNN を用い, モデルの回転にも不変な認識性能をもつ, 境界表現モデル上の形状特徴認識手法を提案する. 本報では回転不変なグラフ上の特徴量構成法と, 形状特徴認識のための GNN 構造を提案する. また, FE メッシュ分割向け形状特徴であるボスやリブ認識用独自データセットや, モデルセグメンテーション用の公開 CAD データセットに対し, 提案手法の認識性能を検証した結果を報告する.

2. GNN を用いた形状特徴認識手法

2.1. 概要

図 1 に提案手法の流れを示す. 学習フェーズでは, まず学習用の境界表現モデルを, 特徴量付き面分隣接グラフへ変換する. 面分隣接グラフとは, 境界表現モデルの面分を節点, 稜線を辺で表現したグラフである. このグラフの節点と辺には, 対応する面分と稜線形状を表す多次元ベクトルの特徴量が付加される. 学習用モデルには, 各面分に形状特徴ラベルが付加されており, グラフ上の対応節点にも, このラベルを付加する. 次に, この特徴量付き面分隣接グラフを訓練データとして用い, グ

ラフ節点のラベリングを行う GNN を学習させる. 一方, 認識フェーズでは, 入力境界表現モデルを同様に特徴量付き面分隣接グラフへ変換し, 学習済み GNN へこのグラフを入力し節点ラベルを出力する. 最後に, この節点ラベルを元の境界表現モデル面分ラベルに反映することで, 形状特徴となる面分を表示する.

2.2. 特徴量付き面分隣接グラフへの変換

本研究では, 面分隣接グラフの節点/辺に付加する特徴量として, 境界表現モデルの Geometry 表現である曲面・曲線の方程式から容易に計算でき, かつモデルの位置・姿勢に不変となる以下の特徴量を設計した.

節点特徴量は, Shape Index 特徴量 F_{SI} と OBB (Oriented Bounding Box) 特徴量 F_{OBB} からなる. F_{SI} は面分上の各サンプリング点 p における Shape Index [4] $SI(p)$ の値の統計分布を表した特徴量である. $SI(p)$ は, 点 p における曲面の主曲率 $\kappa_1, \kappa_2 (\kappa_1 \geq \kappa_2)$ から式 (1) で定義され, 相対変換不変である.

$$SI(p) = (\pi/2) \tan^{-1}((\kappa_2 + \kappa_1)/(\kappa_2 - \kappa_1)) \quad (1)$$

$SI(p)$ は, 曲面の曲率に応じ $-1 \sim 1$ の値をとり, 平面上では値が定義されない. 特徴量 F_{SI} は, サンプル点における $SI(p)$ の正規化ヒストグラムであり, 次式 (2), (3) で定義される.

$$F_{SI} = [|A_0|, |A_1|, \dots, |A_l|]^T / \sum_{i=1}^l |A_i| \quad (2)$$

$$A_i = \begin{cases} \{p \mid -1 + \frac{2i}{l} \leq SI(p) \leq -1 + \frac{2(i+1)}{l}\} & (i \leq l-1) \\ \{p \mid p \text{ is on a plane}\} & (i = l) \end{cases} \quad (3)$$

一方, F_{OBB} は, 面分上のサンプリング点集合に対する OBB の辺比であり, OBB の辺長 $l_1, l_2, l_3 (l_1 \geq l_2 \geq l_3)$ を用いて式 (4) により定義される.

$$F_{OBB} = [l_1, l_2, l_3]^T / \|[l_1, l_2, l_3]^T\| \quad (4)$$

辺特徴量は, 局所的な面分間角度 F_{Ia} と, 大域的な面分間角度 F_{Ga} の2つからなる. F_{Ia} は, ある稜線上のサンプリング点 q において隣接二面分間の符号付き角度 $\theta_{Ia}(q)$ を式 (5), (6) により計算し, これを該当稜線上で平均化することで得られる.

$$F_{Ia} = \frac{1}{|Q|} \sum_{q \in Q} \theta_{Ia}(q) \quad (5)$$

$$\theta_{Ia}(q) = (1/\pi) \operatorname{sgn}((\mathbf{n}_i \times \mathbf{n}_j) \cdot \mathbf{t}_i) \cos^{-1}(\mathbf{n}_i \cdot \mathbf{n}_j) \quad (6)$$

ここで, $Q (= \{q\})$: 稜線上の点 q の集合, $\mathbf{n}_i, \mathbf{n}_j$: 点 q における隣接二面分 f_i, f_j の単位法線ベクトル, \mathbf{t}_i : 面分 f_i に対する半稜線上の点 q における単位接線ベクトル, である.

一方, 大域的な面分間角度 F_{Ga} は, ある稜線の隣接二面分上の平均的な単位法線ベクトル $\bar{\mathbf{n}}_i, \bar{\mathbf{n}}_j$ のなす角を表し, 式 (7) により計算される. ここで $\bar{\mathbf{n}}_i, \bar{\mathbf{n}}_j$ は, 隣接面分 f_i, f_j 上のサンプリング点における単位法線ベクトルを平均化することで求める.

$$F_{Ga} = (1/\pi) \cos^{-1}(\bar{\mathbf{n}}_i \cdot \bar{\mathbf{n}}_j) \quad (7)$$

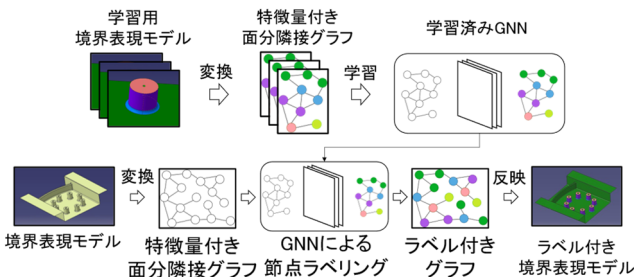


図 1. 提案手法の流れ

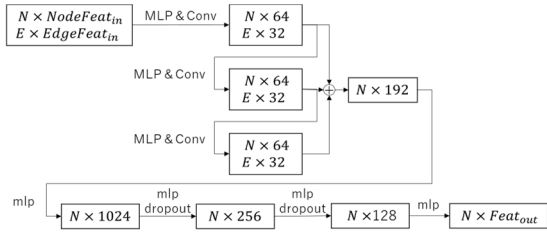


図 2. 提案 GNN のネットワーク構成

2.3. 節点ラベリングのための GNN

節点ラベリングのための GNN のネットワーク構成を図 2 に示す。各矩形内の記号はグラフ上の特徴量サイズを表し、 N は節点数、 E は辺数、 $NodeFeat_{in}$ 、 $NodeFeat_{out}$ は節点の入力特徴量・出力特徴量サイズ、 $EdgeFeat_{in}$ は辺の入力特徴量サイズである。

ネットワークは、前段が畳み込み、後段が多層パーセプトロン (mlp) とドロップアウトからなる節点特徴量に基づく節点ラベリングの機能をもつ。前段の処理は、3 層の MLP&Conv 層からなり、各層は 3 つの処理を行う。まず、辺と節点特徴量の夫々を、mlp やバッチノーマライゼーションレイヤへと入力する。次に、辺特徴量に対し、式 (8) に従って節点特徴量を畳み込む。最後に、計算された辺特徴量を用いて、式 (9) に従い節点特徴量間の畳み込み [5] を行う。ここで、 x_i は節点 i の計算前特徴量、 x'_i は計算後特徴量、 e_{ij} は節点 i と節点 j を結ぶ辺の計算前特徴量、 e'_{ij} はその辺の計算後特徴量、BN はバッチノーマライゼーション、 σ は活性化関数、 \mathcal{N}_i は節点 i 自身とそれに隣接する節点の集合、 \oplus は列ベクトル同士の縦方向結合、 W, W_n, W_e は学習パラメータである重みを表す。

$$e'_{ij} = \sigma \left(\text{BN} \left(W(x_i \oplus x_j \oplus e_{ij}) \right) \right) \quad (8)$$

$$x'_i = \sigma \left(\frac{\exp(\sigma(W_n x_i \oplus W_n x_j \oplus W_e e_{ij}))}{\sum_{k \in \mathcal{N}_i} \exp(\sigma(W_n x_i \oplus W_n x_k \oplus W_e e_{ik}))} W x_j \right) \quad (9)$$

3. 形状特徴認識性能の検証

提案手法を Python と OpenCASCADE により実装し、FE メッシュ分割向け形状特徴であるボス・リブのみを含む独自データセットを用い、認識性能の検証を行った。このデータセットは図 3(a) に示す 17 種のボス・リブ単体モデルに対し、寸法パラメータを変化させ生成したもので、面分に図 3(b) に示す 7 種のラベルが付加されており、合計 33,205 個のモデルを含んでいる。このボス・リブ単体モデルを学習、検証 (validation)、テスト用

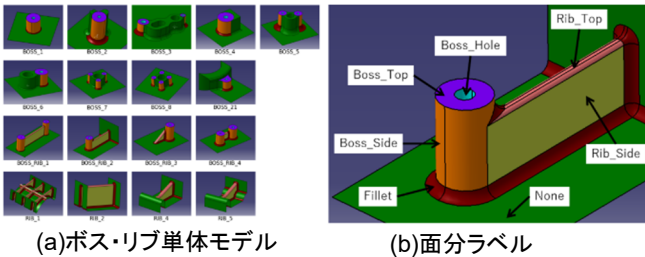


図 3. ボスやリブを含むデータセット

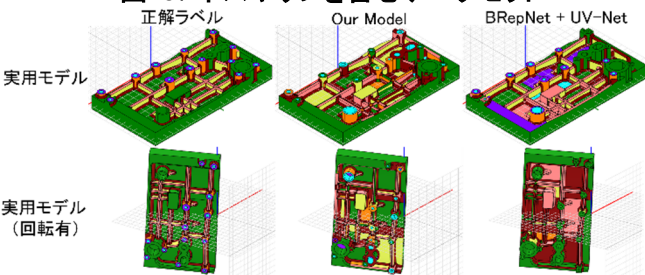


図 4. 実用モデル上の形状特徴認識結果

表 1. ボス・リブデータセットでの認識性能

認識モデル		Accuracy	mIoU
単体モデル	Our Model	0.9965	0.9923
	BRepNet+UV-Net	1.0000	1.0000
単体モデル (回転有)	Our Model	0.9965	0.9923
	BRepNet+UV-Net	0.6270	0.2833
実用モデル	Our Model	0.6920	0.2854
	BRepNet+UV-Net	0.7192	0.4746
実用モデル (回転有)	Our Model	0.6920	0.2861
	BRepNet+UV-Net	0.4809	0.1423

表 2. Fusion360 データセットでの認識性能

認識モデル		Accuracy	mIoU
オリジナル	Our Model	0.853585	0.676659
データセット	BRepNet+UV-Net	0.938108	0.732713
データセット	Our Model	0.846035	0.659667
(回転有)	BRepNet+UV-Net	0.750177	0.428554

のサブセットに様々なパターンで分割し、学習時に検証データセットに対する損失が最小となる認識モデルを用い、認識性能の評価を行った。性能評価は、ボス・リブ単体モデルのデータセット、ならびに図 4 に示す複雑な実用モデルに対して実施し、夫々、元々の姿勢のモデルと、ランダム回転を与えたモデルを入力した。また性能の比較対象として、本提案手法と類似した形状特徴認識用 GNN である BRepNet [2] と UV-Net [3] を組み合わせたネットワーク [6] による認識も行った。その結果を表 1 に示す。また、同様の検証を CAD モデルセグメンテーション用公開データセットの Fusion 360 gallery [7] に含まれる 32,110 個の境界表現モデルに対し行った結果を、表 2 に示す。

ボス・リブ単体モデルでの検証においては、入力モデル姿勢が学習データセットと類似している場合は、本提案手法の認識性能は BRepNet + UV-Net にわずかに及ばなかったが、入力モデルを回転した場合は、より良い性能を示した。また本手法では、モデル回転の有無による認識性能低下はみられず、目標とした回転不変の形状特徴認識が実現されていることがわかる。一方、本手法を用いた FE メッシュ向け実用モデルや Fusion360 モデルに対する認識性能は、BRepNet + UV-Net に比べ mIoU が低く、今後、認識モデルの汎化性能や学習用データセット作成に改善の余地があることが分かった。

4. まとめ

GNN を用いた境界表現モデル上の形状特徴認識のため、モデルの回転に依存しないグラフと特徴量の構成法を提案し、認識性能が回転に影響されないことを示した。しかし、汎化性能を含め、認識性能のさらなる向上が今後必要と考えられる。

参考文献

- [1] Z. Zhang, et al.: "FeatureNet: Machining feature recognition based on 3d convolution neural network," Computer-Aided Design, 101, pp.12–22, 8 (2018).
- [2] J. G. Lambourne, et al.: "Brep-net: A topological message passing system for solid models," Proc. of Computer Vision and Pattern Recognition (CVPR), 6, pp.12773–12782, (2021).
- [3] P. K. Jayaraman, et al.: "UV-net: Learning from boundary representations," Proc. of Computer Vision and Pattern Recognition (CVPR), 6, pp.11703–11712, (2021).
- [4] J. J. Koenderink, et al.: "Surface shape and curvature scales", Image and Vision Computing, 10(8), pp.557-564, (1992).
- [5] P. Veličković, et al.: "Graph attention networks," (2018). [Online]. Available: <https://openreview.net/forum?id=rJXMpikCZ>
- [6] Autodesk AILab, "GitHub - AutodeskAILab/BRepNet: BRepNet: A topological message passing system for solid models", <https://github.com/AutodeskAILab/BRepNet>, 2022/7/28.
- [7] Karl D. D. Willis, et al.: "Fusion 360 gallery: a dataset and environment for programmatic CAD construction from human design sequences", ACM Trans. Graph, 40(4), Article 54 (2021).