Construction of Shape Atlas for Abdominal Organs using Three-Dimensional Mesh Variational Autoencoder

Ryuichi Umehara[1], *Student Member, IEEE*, Mitsuhiro Nakamura[2], and Megumi Nakao[1], *Member, IEEE*

***Abstract*— A model that represents the shapes and positions of organs or skeletal structures with a small number of parameters may be expected to have a wide range of clinical applications, such as radiotherapy and surgical guidance. However, because soft organs vary in shape and position between patients, it is difficult for linear models to reconstruct locally variable shapes, and nonlinear models are prone to overfitting, particularly when the quantity of data is small. The aim of this study was to construct a shape atlas with high accuracy and good generalization performance. We designed a mesh variational autoencoder that can reconstruct both nonlinear shape and position with high accuracy. We validated the trained model for liver meshes of 125 cases, and found that it was possible to reconstruct the positions and shapes with an average accuracy of 4.3 mm for the test data of 19 cases.**

I. INTRODUCTION

A shape atlas of organs is a model that represents geometric information such as the positions and shapes of organs and their interpatient variability with low-dimensional parameters. Shape atlases are widely used in the fields of medical image analysis and biomedical engineering. A variety of atlases have been constructed using the concept of statistical shape model (SSM) [1][2] and have been used in medical image registration [3], surgical planning [4], and prediction of tumor location in radiotherapy.

Methods for constructing an SSM are divided into two categories [5]: image-based methods [6] and mesh-based methods [7]. Image-based methods use medical images directly to construct SSMs in the medical field. In addition, because images consist of structured pixels, it is easy to extract features from them using filtering operations. However, the quantity of data increases in proportion to the cube of the volume size, making it difficult to construct high-resolution models and to represent complex shapes and nonlinear and discontinuous deformations between organs. In contrast, mesh-based methods require preprocessing steps, such as mesh generation from medical images and deformable registration, to obtain point-to-point correspondence [7]. In addition, calculation of feature values is more complex because of the graph structure of meshes. However, compared with images, meshes can represent shapes with a very small amount of data, and they have the feature that deformation is easy to handle. This study employed a mesh-based model and focused on its shape representation and generalization performance.

[1]R. Umehara is with Graduate School of Informatics, Kyoto University, Kyoto, 606-8501, JAPAN. (email: u-ryuichi@sys.i.kyoto-u.ac.jp,)

[2]M. Nakamura and M. Nakao are with Graduate School of Medicine, Kyoto University, Kyoto, 606-8501, JAPAN.

In a mesh-based SSM, features are extracted from a registered organ mesh using principal component analysis (PCA) or independent component analysis (ICA), and the shape variability of organs is expressed by the weighted sum of the extracted features. However, most medical image databases contain only hundreds of samples. In addition, meshes have higher-dimensional data structures of vertices in the three-dimensional (3D) space. Therefore, our method targets high-dimensional and small-sample datasets. In addition, the shapes and positions of organs vary greatly between patients. Because of this variation, it is difficult for linear models such as PCA and ICA to represent large local deformations. Although some studies have employed kernel methods [7], nonlinear models are prone to overfitting to a limited quantity of data.

Recent studies in geometric modeling have focused on deep learning techniques, and a deep-learning-based SSM using an autoencoder (AE) has been reported [8]. Other studies have used a mesh variational autoencoder (VAE) to construct a shape atlas [9][10]. The VAE is widely used for images, and may be expected to achieve higher generalization performance than the AE because it represents input data as a distribution of latent variables in a low-dimensional space. However, these models use deformation-gradient-based local features and do not cover the reconstruction including the absolute coordinate positions of the vertices.

The purpose of this study was to construct a shape atlas for abdominal organs that represents nonlinear shape differences between patients with high interpretability. As a first step toward this goal, we propose a mesh VAE framework that reconstructs both the shapes and positions of individual organs by using registered organ meshes as both input and output. To improve the correctness of the model, we consider additional localized features, which have been used in recent studies. In our experiments, the mesh VAE model was trained using a set of registered liver meshes generated from 3D computerized tomography (CT) images of 125 patients. We evaluated the reconstruction performance of the model and determined the features that are effective for shape reconstruction.

II. METHODS

Fig. 1 shows the overall structure of the proposed mesh VAE model. The model consists of an encoder and a decoder, and uses a graph convolutional network (GCN) for the encoder part. During mesh reconstruction, the feature X_{input}, which is the input to the model, is first calculated from the vertex and edge information of the organ mesh. The

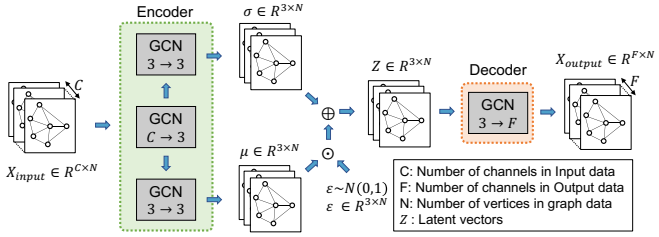

Fig. 1. Proposed model. The encoder learns transformations from inputs to three-channel latent variables, and the decoder learns transformations from latent variables to outputs.

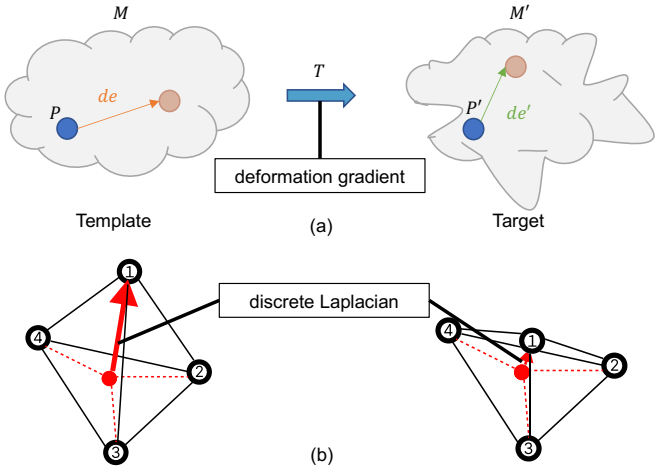

Fig. 2. Concepts of the features used in this paper. (a): deformation gradient. (b): discrete Laplacian feature. Left case in which the curvature is high. Right case in which the curvature is low

encoder then calculates the latent variable Z to be defined for each vertex X by the GCN from the input features, and the decoder reconstructs the features of the mesh X_{output} from the obtained latent variable Z. The learning of weights is accomplished by minimizing the loss function calculated from the inputs and outputs of the model.

A. Organ Mesh Features

Although the vertex positions of a mesh can be used as input features for the proposed model, organs may have complex 3D shapes. Highly accurate shape reconstruction may not always be achieved from vertex positional information alone. Therefore, in this study, we considered the following three types of features and their combinations, which can be calculated from the mesh dataset:

- positions of vertices (three channels),
- deformation gradient (DG) (nine channels), and
- discrete Laplacian (DL) (three channels).

1) Deformation Gradient: The DG is a matrix that describes the change in the positional relationship between corresponding points in two objects, as shown in Fig. 2 (a). Suppose that an object M becomes M' because of local deformation such as stretching, bending, or twisting. Points P and $P + de$, separated by a small distance in M, have moved to P' and $P' + de'$, respectively, because of the deformation. The DG matrix T between P and $P + de$ is defined in (1).

$$de' = Tde \tag{1}$$

T is a 3×3 matrix if the vertex coordinates are 3D vectors. By applying polar decomposition, a type of singular value decomposition, to T, T is decomposed into a positive definite symmetric matrix S and a rectangular matrix R, as follows:

$$T = RS \tag{2}$$

where R is the rotation matrix representing the rotation or bending in the deformation and S is the distortion tensor. For example, if the object M is not deformed at all but only movement occurs, T is the unit matrix I from (1). In parallel translation, no rotation or distortion of the shape occurs, and $R = S = I$ in (2). These equations are consistent with the properties exhibited by the object.

Let M be the shape of the average shape template before mesh deformation positioning. We calculate the DG $T^{(i)}$ that represents the deformation of each case i to the mesh M_i and create the feature $f^{(i)}$.

First, consider the energy $E(T_j^{(i)})$ with DG $T_j^{(i)}$ for vertex j in mesh M_i, defined by

$$E(T_j^{(i)}) = \sum_{k \in N_j} c_{jk} |e'_{jk} - T_j^{(i)} e_{jk}|^2 \tag{3}$$

where

$$e'_{jk} = v'_j - v'_k,$$

$$e_{jk} = v_j - v_k.$$

Here, v'_j denotes the vertex position of M' and v_j denotes the corresponding vertex position of M. N_j denotes the set of vertices adjacent to vertex j, and c_{jk} is defined as follows:

$$c_{jk} = cot\alpha + cot\beta,$$

where α and β refer to the angles located on opposite sides of two triangles that share side e_{jk}. The purpose of coefficient c_{jk} is to maintain the smoothness of the mesh surface in the optimization of (3).

Second, we define the rotational difference as (4).

$$dR_{jk} = R_j^T R_k \tag{4}$$

dR_{jk} represents the rotation between edges that connect adjacent vertices, excluding the rotational component in the entire mesh. Furthermore, by calculating the natural logarithm $ln(dR_{jk})$ of the rotational difference, the value of the element becomes the rotation angle.

Because the stretch tensor S_j is a positive definite target matrix, the deformation-gradient-based feature f_j at vertex j is expressed using three independent components of dR_{jk} and six independent components of S_j, as follows:

$$f_j^M = \frac{1}{|N_j|} (ln(dR_{jk}); S_j)(\forall k \in N_j) \tag{5}$$

2) Discrete Laplacian: The DL is a 3D feature defined at each vertex as the displacement vector between the position of the target vertex and the center position of its surrounding vertices. It is a feature that approximates the average curvature at each vertex for meshes in which the edge length and number of adjacent vertices are relatively uniform across the mesh [11]. As shown in Fig. 2 (b), a larger curvature at each vertex corresponds to a larger DL.

We now explain how the DL is calculated. The DL vector is calculated from the graph Laplacian L, which is defined in (6).

$$L = D - A \tag{6}$$

where D is the degree matrix, which represents the numbers of adjacent vertices, and A is the adjacency matrix, which represents whether pairs of vertices are adjacent. The DL vector for the set of vertices $V \in \{v_i\}$ is defined in (7).

$$f^{DL} = LV \tag{7}$$

For each vertex, this is expressed as follows:

$$f_i^{DL} = |N_i| v_i' - \sum_{j \in N_i} v_j' \tag{8}$$

B. Network Structure

In this study, two types of network configurations were considered: the fully connected (FC) model used in Mesh-VAE [9] and the network configuration using a GCN. A GCN applies convolutional operations, which are widely used in deep learning methods in the image processing field, to graph data. The FC-based model determines output components from all elements of the input vector, whereas the GCN-based model calculates output components from information on adjacent vertices only. Therefore, the GCN-based model is thought to be more efficient for learning.

C. Loss Function

To ensure good generalization performance and highly accurate reconstruction, we introduce a loss function that considers both the distribution of latent variables and input–output reconstruction errors.

For the former, we introduce $\mathcal{L}_{KL}$, defined in (9), to make the distribution of the latent variable Z close to a multiple normal distribution.

$$\mathcal{L}_{KL} = \frac{1}{N} \sum_{i=0}^{N} DK[N(\mu_i, \sigma_i^2) || N(0,1)] \tag{9}$$

where $N(\mu, \sigma^2)$ means the normal distribution. μ and σ are the mean and the standard deviation respectively. For the latter, we introduce the reconstruction loss $\mathcal{L}_F$ for feature F which means the positional feature, the DL feature or the DG feature, as defined in (10).

$$\mathcal{L}_F = MSE(F^{label}, F^{pred}) \tag{10}$$

The overall loss function $\mathcal{L}$ is defined in (11).

$$\mathcal{L} = \mathcal{L}_{KL} + \alpha \mathcal{L}_{Pos} + \beta \mathcal{L}_{DG} + \gamma \mathcal{L}_{DL}, \tag{11}$$

where $\mathcal{L}_{Pos}$, $\mathcal{L}_{DG}$, and $\mathcal{L}_{DL}$ denote the reconstruction errors for vertex position, DG, and DL, respectively. However, (11) defines the loss function when all features are used, and the coefficients are set to 0 for features that are not used in a specific experiment. For example, if only vertex position and DG are used as features, we set $\gamma = 0$.

III. EXPERIMENTS AND RESULTS

To confirm the effectiveness of the proposed method, experiments were conducted on several combinations of the proposed features and models. Python 3.9, PyTorch, and PyTorch-Geometric were used to implement the proposed model. The batch size for training was set to 32, the maximum number of training epochs was set to 1000, and EarlyStopping was used to monitor the $\mathcal{L}_{Pos}$ of the validation data. The EarlyStopping patience was set to 50 epochs. The network was optimized using Adam with a learning rate set to 10^{-2}. For the hyperparameters of the loss function, we checked the performance of the model for several combinations and decided to use the combination with the best performance: $\alpha = 10^{12}$, $\beta = 10^{12}$, and $\gamma = 10^{6}$.

A. Dataset and Preprocessing

Experiments were conducted on liver meshes obtained from abdominal 3D-CT data of 125 patients undergoing radiotherapy for pancreatic cancer at the Department of Radiotherapy, Kyoto University Hospital. After obtaining surface meshes for each organ region from its 3D contours, defined by a radiation oncologist, a liver mesh was obtained by template-based deformable mesh registration. These preprocessing procedures were the same as those used in our previous studies[11]. The overall experiments were approved by the Kyoto University Medical Ethics Committee (approval number: R1446).

The vertices of the meshes were located in the range of $[-256, 256]$ (mm) in coordinates with the origin at the position of the pancreatic cancer: the target of the therapeutic beam in radiation therapy. We normalized the range of feature values to $[-1, 1]$. The 125 meshes used in the experiment were split into three sets: 87 for training, 19 for validation, and 19 for testing.

B. Evaluation of Mesh Reconstruction Performance

In this experiment, to explore the combinations of features and models that are useful for reconstructing organ meshes, we trained models to compare the combinations listed in Table I and obtained reconstruction results for the liver mesh. We also evaluated a PCA-based algorithm with six eigenvectors as a conventional linear shape estimation approach. For quantitative evaluation of the results, the distance error E between the vertices of the reconstruction result v' and the corresponding vertices of the target v was calculated using (12) for each dataset.

$$E = \frac{1}{NM} \sum_{i}^{M} \sum_{j}^{N} ||v_i^{(j)} - v_i'^{(j)}||_2 \tag{12}$$

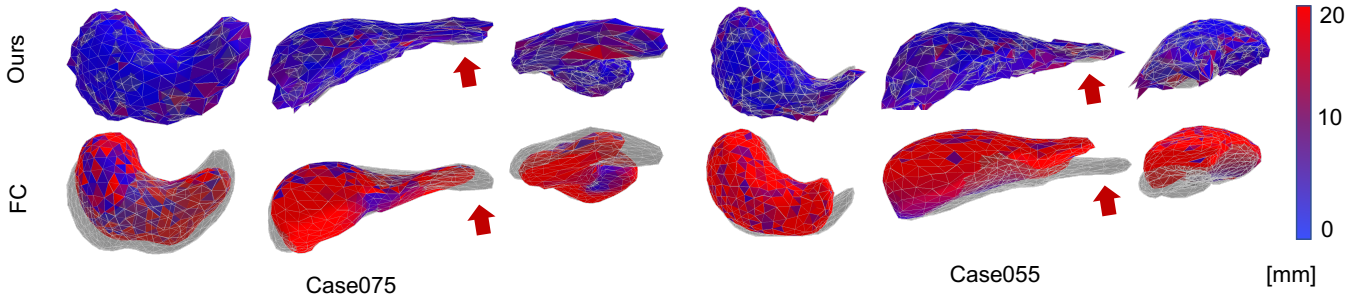

Fig. 3. Reconstruction results for the two cases with the largest reconstruction errors in the test data. From left to right: axial, coronal, and sagittal. The gray area depicts the mesh of the label data, and the colored mesh depicts the reconstruction results. The red area indicates a larger error, and the blue area indicates a smaller error. [Ours] feature: Position + DL + DG, model: GCN-based, [FC] feature: Position, model: FC-based

TABLE I

MODEL AND FEATURE COMBINATIONS IN TRAINING RESULTS. THE EPOCH COLUMN SHOWS THE NUMBER OF EPOCHS WITH THE BEST RECONSTRUCTION ERROR ON THE VALIDATION DATA, AS DETERMINED BY EARLYSTOPPING. THE COLUMNS TRAIN, VAL, AND TEST SHOW THE DISTANCE ERROR. THE UNIT OF DISTANCE IS MM

model	feature	epoch	Train	Val	Test
PCA	Position	-	5.4927	5.3511	5.1410
FC	Position	230	6.7509	11.929	12.158
GCN	Position	995	5.3704	5.4113	5.3337
GCN	Position	995	5.3704	5.4113	5.3337
GCN	Position + DG	676	5.1983	5.1691	5.2139
GCN	Position + DL	981	4.2451	4.2300	4.3023
GCN	**Position + DL + DG**	**991**	**4.2126**	**4.2298**	**4.2643**

where N is the number of vertices in one mesh and M is the number of mesh in a dataset. From Table I, it can be observed that the GCN-based model using the vertex position + DL + DG features had the lowest error on the test data. The GCN-based model using the vertex position + DL features had the second-lowest error. To compare these two models, a two-tailed Wilcoxon signed rank test was conducted at a significance level of 0.05. The p-value was 0.007145, confirming that the results of the two models were significantly different.

Two examples of the reconstruction results are visualized in Fig. 3. The figure shows that the FC-based model had a large overall error, whereas the GCN-based model was able to reconstruct the entire image with high accuracy. In particular, at the tip of the left lobe (indicated by the arrow in the figure), the FC-based model was far from the correct position, whereas the GCN-based model was able to reconstruct the image with high accuracy. This confirms that the GCN-based model is superior to the FC-based one.

IV. CONCLUSIONS

In this study, we designed a mesh VAE that can reconstruct both the shape and position of individual organs with high accuracy using 3D meshes as input and output. This achieved our objective, namely to construct a shape model with high accuracy and high generalization performance. The model was validated using organ meshes from 125 cases, and the positions and shapes were reconstructed with an average accuracy of 4.3 mm. For future work, guided by the results obtained in this study, we will revise the architecture of the model to improve its interpretability, strive to elucidate the low-dimensional parameters that constitute organ shapes, and apply the method to other abdominal organs.

ACKNOWLEDGMENT

This research was supported by JSPS Grant-in-Aid for Scientific Research (B) (grant number 22H03021 and 19H04484).

REFERENCES

[1] F. Stefan, B. Mario, "Example-driven deformations based on discrete shells," Computer Graphics Forum, vol.30, pp. 2246-2257, 2011.
[2] R. W. Sumner, M. Zwicker, C. Gotsman, J. Popović, "Mesh-based inverse kinematics," Special Interest Group on Computer Graphics (SIGGRAPH), pp. 488–495, 2005.
[3] B. Koo, E. Özgür, B. Le Roy, E. Buc, A. Bartoli, "Deformable registration of a preoperative 3D liver volume to a laparoscopy image using contour and shading cues," in Medical Image Computing and Computer Assisted Intervention (MICCAI), pp. 326–334, 2017.
[4] S. Raith et al., "Planning of mandibular reconstructions based on statistical shape models," International Journal of Computer Assisted Radiology and Surgery, vol. 12, no. 1, pp. 99–112, 2017.
[5] T. Heimann. H. Meinzer, "Statistical shape models for 3D medical image segmentation: A review," Medical Image Analysis, vol. 13, no. 4, pp. 543–563, 2009.
[6] S. Oh, S. Kim, "Deformable image registration in radiation therapy," Radiation Oncology Journal, vol. 35, no. 2, pp. 101–111, 2017.
[7] M. Nakao, M. Nakamura, T. Mizowaki, T. Matsuda, "Statistical deformation reconstruction using multi-organ shape features for pancreatic cancer localization", Medical Image Analysis, vol. 67, p. 101829, 2021.
[8] H. Dai, L. Shao, "PointAE: point auto-encoder for 3D statistical shape and texture modelling," International Conference on Computer Vision (ICCV), pp. 5409-5418, 2019.
[9] Q. Tan, L. Gao, Y. K. Lai, S. Xia, "Variational autoencoders for deforming 3D mesh models," Computer Vision and Pattern Recognition (CVPR), pp. 5841-5850, 2018.
[10] Y. J. Yuan, Y. K. Lai, J. Yang, Q. Duan, H. Fu, L. Gao, "Mesh variational autoencoders with edge contraction pooling," Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 1105-1112, 2020.
[11] M. Nakao, M. Nakamura, T. Matsuda, "Image-to-graph convolutional network for 2D/3D deformable model registration of low-contrast organs," IEEE Trans. on Medical Imaging, Vol. 41, No. 12, pp. 3747-3761, 2022.