Knowledge-Based Systems 236 (2022) 107360

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Dynamic graph convolutional autoencoder with node-attribute-wise attention for kidney and tumor segmentation from CT volumes

Ping Xuan [a], Hui Cui [b], Hongda Zhang [a,*], Tiangang Zhang [c,*], Linlin Wang [d], Toshiya Nakaguchi [e], Henry B.L. Duh [b]

[a] School of Computer Science and Technology, Heilongjiang University, Harbin, China
[b] Department of Computer Science and Information Technology, La Trobe University, Melbourne, Australia
[c] School of Mathematical Science, Heilongjiang University, Harbin, China
[d] Department of Radiation Oncology, Shandong Cancer Hospital and Institute, Shandong First Medical University and Shandong Academy of Medical Sciences, Jinan, China
[e] Center for Frontier Medical Engineering, Chiba University, Chiba, Japan

ARTICLE INFO

Article history:
Received 3 March 2021
Received in revised form 1 June 2021
Accepted 1 August 2021
Available online 4 August 2021

Keywords:
Kidney and tumor segmentation
Graph node attributes
Dynamic graph convolutional autoencoder
Node-attribute-wise attention
Long-distance relationship between nodes

ABSTRACT

Extraction and integration of semantic connections, spatial relations and dependencies are critical in volumetric image segmentation. This is a challenging issue, especially when there are long-distance objects with close semantic relations and neighboring objects with indistinct boundaries. We propose a novel dynamic graph convolution (DGC) autoencoder with node-attribute-wise attention (NodeAttri-Attention) for relation inference and reasoning, with applications on kidney and tumor segmentation from computerized tomography (CT) volumes. We first introduce a new graph construction strategy for 3D volumetric image data, where graph node attributes and connections represent topological relations and high-level correlations. Then NodeAttri-Attention mechanism is proposed to obtain attention-enhanced node attributes by discriminating adaptive contributions of various features. Finally, the DGC strategy is designed to learn and integrate the complex and underlying correlations across image regions. Our DGC dynamically updates graph topology and node attributes as the graph convolutional layer gradually deepens. Experimental results and ablation studies demonstrated the effectiveness of each of our major innovations in NodeAttri-Attention DGC, especially when objects are with weak boundaries, irregular shapes, and various sizes. The improved segmentation results of embedding NodeAttri-Attention DGC to different segmentation backbones show the generality of DGC autoencoder.

1. Introduction

Segmentation of regions of interests (ROIs) from computed tomography (CT) volumes is an essential step in the clinical routine for cancer diagnosis and treatment planning. Manual delineation of ROI is time consuming and requires domain knowledge. Recently, automated image segmentation models demonstrate competitive results by learning from large number of images and ground truth (GT) labels [1]. However, accurate segmentation of kidney and tumor from CT images is a challenging task [2,3]. This is because objects with the same semantic information are located far away from each other in the images. In this meanwhile, nearby objects with different class labels are with similar intensity distributions. For instance, the semantic and texture features of left and right kidneys are similar to each other when compared with other tissues in abdomen. Kidney tumors with indistinct boundaries exhibit inside the kidney or near the kidney boundary. Since there are strong connections between the locations of tumor and kidneys, effective modeling of the long-distance semantic connections and spatial relations between objects would contribute to the detection and segmentation in CT volume.

Convolutional neural network (CNN) has been used in various image segmentation tasks [4–7] including kidney and tumor segmentations [8,9]. Most of existing models, however, focus on capturing local information, which makes it difficult to preserve the contextual connections between regions located far away from each other. To address this issue, there are mainly two categories of approaches. One is by introducing the global context of the image information into the convolutional layers. A couple of methods [10,11] use multi-scale spatial features to enlarge receptive fields. For instance, deformable convolution network [12] enlarges the receptive field by adding offsets to convolutional sampling. To transfer knowledge from one region to

* Corresponding authors.
E-mail addresses: hongda.zhang.hlju@gmail.com (H. Zhang), zhang@hlju.edu.cn (T. Zhang).

https://doi.org/10.1016/j.knosys.2021.107360
0950-7051/

another located far away, non-local network [13], self-attention mechanism [14], double attention networks [15], and dual attention network [16] have been explored. Recently, recurrent neural networks [17,18] are used to extract and integrate contextual information. The calculations involved in those methods, however, reply on the stacking of large number of convolutions, which makes such approach computational expensive. Besides, the effective modeling of semantic and spatial connections between long-distance image regions cannot be guaranteed.

The second type of approach is based on graph propagation. Graph structures can embed topology and node attributes across different long-distance and disjoint regions in images. For instance, some methods [19–22] firstly project features from the original coordinate space to interaction space for fully connected graph construction. Then graph reasoning is performed based on convolutional networks [23]. Finally, the graph node attributes are projected back to the original coordinate feature space. During the projection and re-projection process, however, detailed information in the original space such as spatial relation cannot be well preserved. To address this issue, a recent method [24] was proposed to get rid of the projection and re-projection procedure by performing graph reasoning in the original feature space directly. However, this method reduced the dimension of the CNN features and performed global pooling before topology extraction. Thus, the high-level representations and detailed features are lost in topology estimation.

To address these limitations, we propose a novel dynamic graph convolution (DGC) autoencoder for relation inference and reasoning with applications on semantic image segmentation. Our contributions are summarized below.

(1) We propose a new graph construction strategy for 3D volumetric image data. The graph node attributes and connections represent topological relations and high-level correlations, aiming to facilitate effective subsequent graph reasoning.
(2) Since the attributes of a node in the graph have different contributions to the representation learning of the long-distance relationship between nodes, we propose a node-wise attribute attention mechanism. The attention mechanism can measure the various contributions of features for each node, and assigns adaptive weights to each of the node attribute vector values to produce informative attributes.
(3) The proposed DGC based encoder and decoder can learn and integrate the complex and underlying correlations across image regions. Compared with static graphs, our graph topology and node attributes dynamically evolve as the encoding layer gradually deepens. The dynamic evolution contributes to the reasoning of semantic information and spatial dependencies between image regions, especially those with long distances. The experimental results demonstrate the effectiveness and generality of each of our major innovations on kidney and tumor segmentation from 3D CT volumes.

2. Related work

Global context-integrated modeling. To alleviate the shortcoming of focusing on local regions by convolution operations, some methods are proposed to learn and integrate global context information. For instance, PSPNet [10] and DenseASPP [11] are proposed to enlarge the receptive field of convolution by enlarging the spatial sampling range. By such, a broader range of context information can be extracted for semantic segmentation. Dai et al. [12] proposed a deformable convolutional network to learn the offsets of convolutional sampling locations and context of broader fields. Squeeze-and-Excitation networks [25] use global average pooling and full connections to encode global contextual information. Non-local network [13], self-attention mechanism [14], double attention networks [15], and dual attention network [16] calculate the correlations between two locations in the feature map or the correlations between channels to capture global dependency. Recently, a recurrent neural networks based model [17,18] is proposed to aggregate the context over locally connected feature maps.

Graph-based long-range relationship reasoning. Graph models have unique features such as node connections and topology relations for node-wise relation reasoning. Graph convolutional neural network (GCN) associates deep topological structure and node attribute information in graphs. [26,27] For instance, GCN is used for node-wise classification [21,24], link prediction [28,29], graph classification [30] and so on. Since information can be propagated along graph edges across different nodes, graph reasoning can also be used in image recognition tasks [19–22]. For instance, random walk (RW) algorithm [31] and conditional random field (CRF) [32] have been used to integrate relations across graph nodes for semantic image segmentation [33,34]. A common approach is to project image features to a new feature space with fewer dimensions and dense representations. Then graphs are constructed with fully connected nodes in the new feature space for node-wise information propagation and relational reasoning. Finally, the relation-aware features are projected back to the original coordinate space for further tasks. The projection operation, however, destroys the spatial relations between image regions. Li et al. [24] proposed to perform feature dimension deduction directly in the original coordinate space, followed by GCN reasoning. Dimension deduction, however, will result in information loss, especially detailed information. Unlike existing approaches, we propose dynamic graph convolutional encoding and decoding strategies to preserve both the spatial relations and the detailed information of nodes. Besides, with the automated adjustment of node-wise information and topological structures, our model can dynamically capture multi-scale relations between long-distance objects.

Unlike existing approaches, we propose dynamic graph convolutional encoding and decoding strategies to preserve both the spatial relations and the detailed information of nodes. Besides, with the automated adjustment of node-wise information and topological structures, our model can dynamically capture multi-scale relations between long-distance objects.

Kidney and tumor segmentation from CT. Automated kidney and tumor segmentation from CT volumes is a challenging issue due to the various size, irregular shape, and blurry boundaries among different patient scans. Yang et al. [35] proposed a 3D fully convolutional neural network with pyramid pooling. However, this method requires the pre-identified region of interest (ROI), which cannot be applied directly to raw CT data. A 3D U-net-based boundary-aware network [36] was proposed to integrate kidney and tumor boundary information. Fabian et al. [36] developed a nnU-Net model which optimized data pre-processing, network architecture, training and post-processing. nnU-Net was used as the basic framework by a few methods and achieved outstanding performance on 2019 kidney and tumor segmentation competition. For instance, MSS U-net [8], an nnU-net based model, was proposed by introducing multi-scale supervision scheme. The long-range spatial dependency between objects such as two kidneys and tumor, however, was neglected. Thus, we propose a dynamic graph convolutional network with attention to model long-range dependencies to improve the performance.

We propose a DGC with attention to model long-range dependencies to improve the segmentation performance. In our

model, deep semantic information and spatial structure are firstly extracted from 3D images by CNN segmentation backbone encoder. To enhance the modeling of long-range semantic dependencies between objects such as left and right kidneys and tumor, we propose dynamic DGC. This is because GCN can propagate the encoded semantic information along graph edges across different nodes, which is not limited by rigid-like feature space. Furthermore, compared with static GCN, our DGC automatically adjust node-wise information and topological structures during the training process.

3. Method

The proposed dynamic graph convolution segmentation model (DGC-Seg) is given in Fig. 1. DGC-Seg consists of three major learning components. The first component is to learn context representations, including texture and semantic features by segmentation backbone. We use 3D nnU-Net [37] as the backbone. Secondly, given the features extracted by segmentation encoder, we design a graph construction strategy to associate image features with graph nodes and topology. Besides, a node-attribute-wise attention mechanism is proposed to generate weighted node attributes. Thirdly, dynamic graph convolution (DGC) based encoder (DGC-Encoder) and decoder (DGC-Decoder) are proposed for dynamic relation reasoning of complex connections between image regions, especially spatial dependencies and semantic relations. Finally, region relation representations and context representations are adaptively integrated and decoded for segmentation output.

3.1. Learning context representations

Given 3D nnU-Net as the backbone segmentation architecture to extract context representation, the 3D encoder has six encoding layers and decoder is composed of six decoding layers. All the encoding layers are composed of $3 \times 3 \times 3$ convolutional block, followed by ins_norm and Leaky ReLU activation. We use stride convolution instead of pooling layer to obtain more accurate and representative context features in each down-sampling stage. For decoding layers, we perform upsampling based on transposed convolution. Skip connections from encoding layer, and decoding layers are used to integrate detailed features of encoder and decoder. Let $\mathbf{F} \in \mathbb{R}^{H\times W\times D\times C}$ denotes the output features of 3D encoder, where H, W, D denotes the height, width and depth, and C denotes the number of channels. $\mathbf{F}$ is considered as the context representation $\mathbf{R}_c$ of input images.

For tasks such as kidney and kidney tumor segmentation, the left and right kidneys have the same semantic meanings. Besides, kidney tumors have strong correlations with the locations of kidneys. Thus, we propose DGC autoencoder to extract the long-range semantic meanings and spatial correlations. In the following sections, we introduce the proposed graph construction method, followed by the DGC strategy.

3.2. Graph with node-attribute-wise attention

3.2.1. Graph construction

Definition 1 (*Graph Node and Attributes*)**.** As shown in Fig. 2, given output feature matrix $\mathbf{F} \in \mathbb{R}^{H\times W\times D\times C}$ from 3D encoder, we construct a graph $G = (V, E, \mathbf{X}, \mathbf{A}_1)$ to represent the correlations between different image regions, where V and E denote graph nodes and edges, $\mathbf{X}$ is the node attribute matrix, and $\mathbf{A}_1$ is the adjacency matrix. As the ith position of $\mathbf{F}$, $\mathbf{F}_i \in \mathbb{R}^{H\times W\times D}$, corresponds to a particular region in the input image, we use a graph node v_i to represent the region and there are $N_V = H \times W \times D$ nodes. For node v_i, its node attribute vector is formed by $[\mathbf{F}_{i1}, \ldots, \mathbf{F}_{iC}]$ and denoted by $\mathbf{X}_i$. $\mathbf{X}_i$ is the ith row of $\mathbf{X} \in \mathbb{R}^{N_V\times C}$.

Definition 2 (*Graph Topology and Adjacency Matrix Initialization*)**.** Given node attribute vectors $\mathbf{X}_i$ and $\mathbf{X}_j$ belonging to nodes v_i and v_j, if the distributions of $\mathbf{X}_i$ and $\mathbf{X}_j$ are more similar, there is higher probability that nodes v_i and v_j are related with each other. We use L_1 distance to measure the similarity between $\mathbf{X}_i$ and $\mathbf{X}_j$ as

$$\varphi_{ij} = \exp\{-\|\mathbf{X}_i - \mathbf{X}_j\|_1\} \tag{1}$$

where exponential operation exp is to rescale the similarities to [0, 1]. φ_{ij} is considered as the edge weight of $e_{ij} \in E$ connecting nodes v_i and v_j. If two nodes are more (less) similar, their edge weight is larger (smaller) and closer to 1 (0). We further perform Laplace normalization on $\mathbf{\Phi} = [\varphi_{ij}] \in \mathbb{R}^{N_V\times N_V}$, and obtain adjacency matrix as

$$\mathbf{A}_1 = \mathbf{D}_1^{-\frac{1}{2}}\mathbf{\Phi}\mathbf{D}_1^{-\frac{1}{2}} \tag{2}$$

where $\mathbf{D}_1$ is a diagonal matrix and $(\mathbf{D}_1)_{ii} = \sum_j \mathbf{\Phi}_{ij}$. Adjacency matrix reflects the edge connection and graph topology. In our dynamic graph, the adjacency matrix is initialized by Eq. (2) and further evolved and learnt during the training process by DGC encoding layers. We detail the dynamic evolution in Section 3.3.1.

For a graph node, different attributes in its attribute vector are from different channels of $\mathbf{F}$. Since different channels may have various importance in decision-making, the node attribute vector values should have different weighted contributions to the learning of relation representation of the node. Thus, we propose a new node-attribute-wise attention to represent node attributes with attentional weights.

3.2.2. NodeAttri-Attention mechanism

Considering channel-wise variations and contextual relations between the attribute vectors of multiple nodes, our attention mechanism is defined as: given $(\mathbf{X}_i)^T$ denoting the transposed ith row of $\mathbf{X}$, the attentional weight vector of node v_i is obtained as

$$u_i = \mathbf{H}_a \tanh(\mathbf{W}_a(\mathbf{X}_i)^T + \mathbf{b}_a) \tag{3}$$

where $\mathbf{H}_a$ is attention weight matrix reflecting contextual relations between graph nodes. $\mathbf{W}_a$ is weight matrix and $\mathbf{b}_a$ is the bias vector. $\mathbf{H}_a$, $\mathbf{W}_a$ and $\mathbf{b}_a$ are randomly initialized, and automatically learnt during the training process.

The normalized attention weight α_{it} of tth attribute of v_i is defined as:

$$\alpha_{it} = \frac{\exp(u_{it})}{\sum_t \exp(u_{it})}, t = 1, \ldots, C \tag{4}$$

Thus, the node-attribute-wise attention enhanced attribute vector $\widetilde{\mathbf{X}}_i$ of $\mathbf{X}_i$ is:

$$\widetilde{\mathbf{X}}_i = \alpha_i \otimes \mathbf{X}_i + \mathbf{X}_i \tag{5}$$

where $\alpha_i = [\alpha_{it}]_{t=1,\ldots,C}$, and $\otimes$ denotes element-wise multiplication.

3.3. DGC autoencoder

In conventional graph convolution, the adjacency matrix $\mathbf{A}_1$ remains unchanged during the encoding process. The graph connection and topology, however, may change with the learning and convolution process. Thus, we propose a dynamic graph convolution mechanism to capture the graph evolution.

3.3.1. DGC-encoder

Given the attentive attribute matrix $\widetilde{\mathbf{X}}$ and adjacency matrix $\mathbf{A}_1$, the first encoding layer is performed as

$$\mathbf{Y}_1 = f_{LeakyReLU}(\mathbf{A}_1\widetilde{\mathbf{X}}\mathbf{W}_1) \tag{6}$$

where $f_{LeakyReLU}$ is Leaky ReLU, and $\mathbf{W}_1$ is weight matrix. $\mathbf{Y}_1$ is the output feature matrix in first layer and it also contains the learned

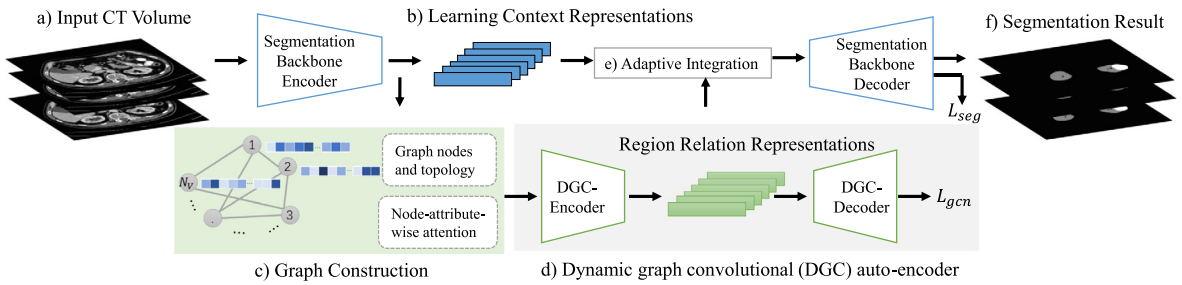

Fig. 1. Overview of the proposed DGC-Seg model. Given input CT volumes, we firstly extract (b) context representations from segmentation backbone encoder. Secondly, (c) a new graph with node-attribute-wise attention is proposed to associate context representation for (d) dynamic graph convolutional (DGC) based encoder and decoder construction. Region relation representations extracted by DGC and context representations are fused by (e) an adaptive integration module before sending to the segmentation decoder. The detailed architecture of graph construction and DGC are given in Figs. 2 and 3.

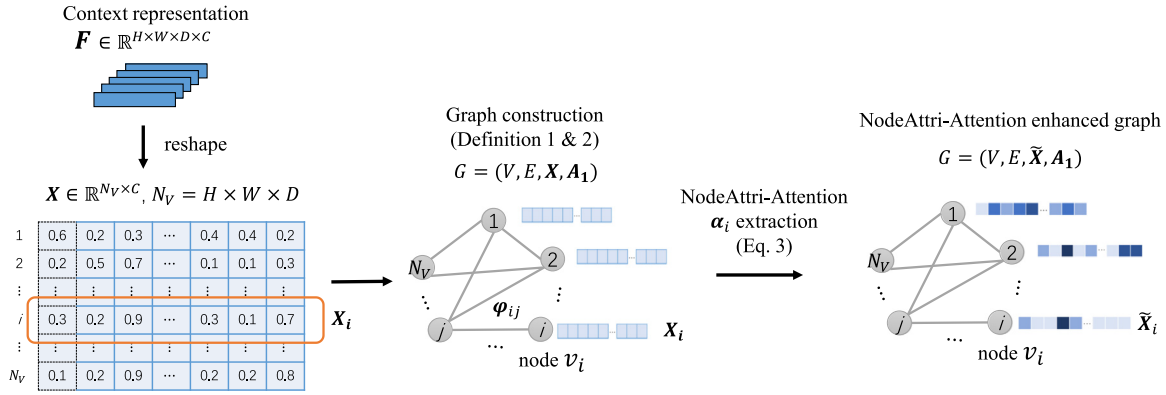

Fig. 2. Illustration of the proposed Graph construction process. The output of segmentation backbone encoder, **F**, is firstly reshaped to $\mathbf{X} \in \mathbb{R}^{N_V\times C}$ where each row corresponds to a graph node attribute vector. Graph edge connections are obtained based on the similarities between two graph nodes. A new node-attribute-wise attention (NodeAttri-Attention) mechanism is proposed to learn various importance of node attributes.

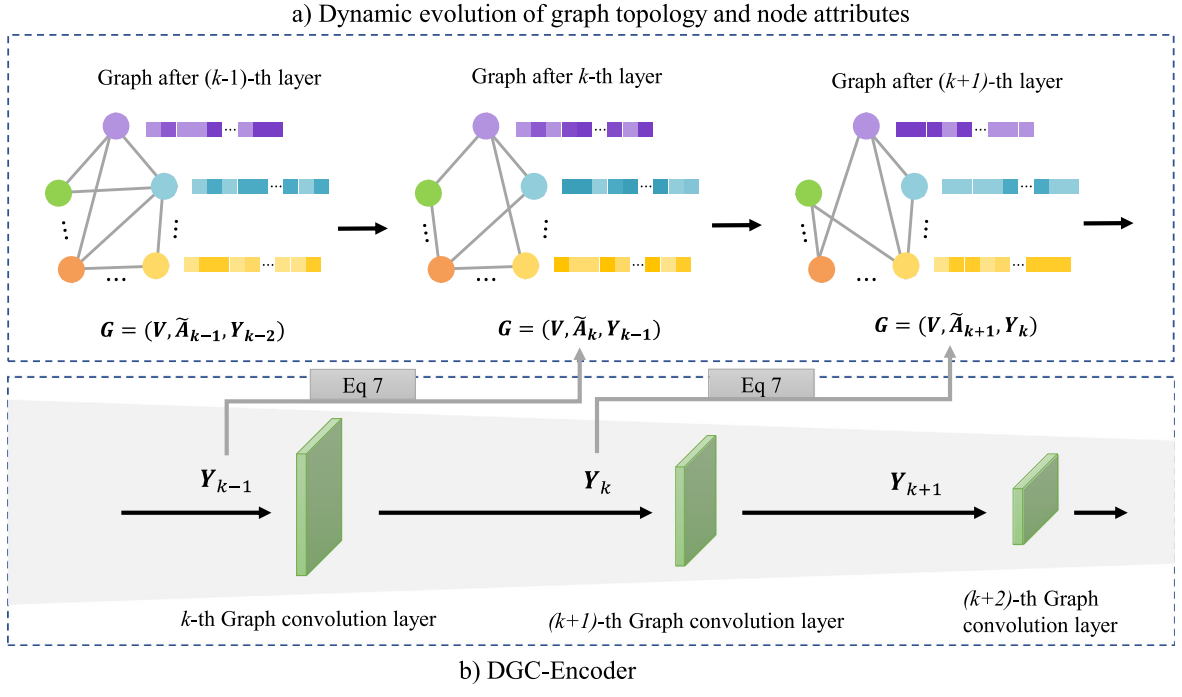

Fig. 3. Dynamic evolution of graph topology in DGC encoder. The adjacency matrix of kth graph is dynamically updated using the output feature from $(k-1)$-th graph convolution layer.

representative features of each node. The ith column in $\mathbf{W}_1$ is considered as the ith filtering window in graph convolution. $\mathbf{W}_1$ is to be learnt during the training process. Afterwards, for each of the following encoding layers, we dynamically update the graph's topological structure.

Dynamic evolution of graph topology. Given the output feature map from $(k-1)$-th graph convolution layer, which is denoted by $\mathbf{Y}_{k-1}$, we first concatenate $\mathbf{Y}_{k-1}$ and node attribute matrix $\widetilde{\mathbf{X}}$ as $\widetilde{\mathbf{Y}}_{k-1}$. By such, both original attribute details and new representative features of nodes are considered during dynamic graph topology evolution process. Secondly, the evolved adjacency matrix $\widetilde{\mathbf{A}}$ is obtained as

$$\widetilde{\mathbf{A}}_k = \widetilde{\mathbf{D}}_k^{-\frac{1}{2}} \Phi_k \widetilde{\mathbf{D}}_k^{-\frac{1}{2}}, \ \widetilde{\Phi}_{k_{ij}} = \exp\{-\|\widetilde{\mathbf{Y}}_{k-1i} - \widetilde{\mathbf{Y}}_{k-1j}\|_1\} \tag{7}$$

where $(\widetilde{\mathbf{D}}_k)_{ii} = \sum_j \widetilde{\Phi}_{k_{ij}}$. The dynamic output of kth layer is obtained as:

$$\mathbf{Y}_k = f_{LeakyReLU}(\widetilde{\mathbf{A}}_k \mathbf{Y}_{k-1} \mathbf{W}_k), k = 2, \ldots, L_e \tag{8}$$

where L_e is number of the encoding layers.

3.3.2. DGC-decoder

The output attribute representation matrix of the last encoding layer, $\mathbf{Y}_{L_e} \in \mathbb{R}^{N_V \times N_f}$, is fed into the first layer of DGC-Decoder as:

$$\widehat{\mathbf{Y}}_1 = f_{LeakyReLU}(\mathbf{A}_1 \mathbf{Y}_{L_e} \widehat{\mathbf{W}}_1) \tag{9}$$

The output of the following DGC decoding layers, $\widetilde{\mathbf{Y}}_k$, is obtained as

$$\widehat{\mathbf{Y}}_k = \begin{cases} f_{LeakyReLU}(\mathbf{A}_1 \widehat{\mathbf{Y}}_{k-1} \widehat{\mathbf{W}}_k), & k = 2, \ldots, L_{d-1} \\ f_{Sigmoid}(\mathbf{A}_1 \widehat{\mathbf{Y}}_{k-1} \widehat{\mathbf{W}}_k), & k = L_d \end{cases} \tag{10}$$

where L_d is the number of decoding layers. As the input of encoder $\widetilde{\mathbf{X}}$ needs to be as consistent as with the output of the decoder $\widehat{\mathbf{Y}}_{L_d}$, the goal of DGC autoencoder optimization is to minimize the following loss function,

$$L_{dgc} = \|\widetilde{\mathbf{X}} - \widehat{\mathbf{Y}}_{L_d}\|_1 \tag{11}$$

3.4. Context and relation representation integration and optimization

Let $\mathbf{Y}_{L_e} \in \mathbb{R}^{N_v \times N_f}$ denote the output from the last encoding layer where N_f is the number of node features, the ith row of $\mathbf{Y}_{L_e}$, $(\mathbf{Y}_{L_e})_i$, is regarded as the region relation representation of ith node. We use $\mathbf{R}_r$ to represent $\mathbf{Y}_{L_e}$ in the following paragraphs. Meanwhile, feature map $\mathbf{F} \in \mathbb{R}^{H \times W \times D \times C}$ can be obtained by nnU-Net encoder. We introduce context representation matrix $\mathbf{R}_c$ by reshaping $\mathbf{F}$ to a matrix of size $N_v \times C$ where $N_v = H \times W \times D$. $(\mathbf{R}_c)_i$, the ith row of $\mathbf{R}_c \in \mathbb{R}^{N_v \times C}$ is the context representation of ith node. As the features in $\mathbf{R}_r$ and $\mathbf{R}_c$ have various contributions to the segmentation results, we perform N_{conv} 1×1 convolutional operation on $\mathbf{R}_r$ and $\mathbf{R}_c$ to automatically learn the adaptive weights. Let w_k denote the kth 1×1 convolutional kernel, the jth value in w_k represents weight of jth feature. Thus, the kth feature by weighted integration is obtained as

$$(\mathbf{z}_i)_k = w_k * [(\mathbf{R}_r)_i, (\mathbf{R}_c)_i], \tag{12}$$

where $*$ denotes convolutional operation, $[(\mathbf{R}_r)_i, (\mathbf{R}_c)_i]$ denotes concatenation operation on $(\mathbf{R}_r)_i$ and $(\mathbf{R}_c)_i$. The integrated matrix $\mathbf{Z}$ is then obtained as $\mathbf{Z} = [(\mathbf{z}_i)_1, (\mathbf{z}_i)_2, \ldots, (\mathbf{z}_i)_{N_{conv}}]$, and reshaped back to $H \times W \times D \times N_{conv}$. The final segmentation result $\mathbf{H}_p$ can be obtained by sending $\mathbf{Z}$ to the corresponding decoding layers in nnU-Net. In each image, the background region usually takes the largest proportion. The foreground kidney tissue occupies a small proportion while the tumor is even smaller. As there are imbalanced class distributions in the segmentation task, we use multi-class cross-entropy loss and Dice loss by following previous work [38]. The segmentation loss L_{seg} is defined as

$$\begin{aligned} L_{seg} &= L_{CE} + L_{Dice} \\ &= -\sum_{c \in C} \sum_{j=1}^{N_o} h_j^c \log p_j^c + \sum_{c \in C} \left(1 - \frac{2 \sum_{j=1}^{N_o} h_j^c \widehat{h}_j^c}{\sum_{j=1}^{N_o} ((h_j^c)^2 + (\widehat{h}_j^c)^2)}\right) \end{aligned} \tag{13}$$

where h_j^c, p_j^c, and $\widehat{h}_j^c$ represent ground truth, predicted probability, and one-hot output of voxel j for the cth class. N_o is the total number of voxels, and C is the number of classes. In this work, our experimental results show that cross-entropy loss and Dice loss can be combined directly without tuning their weights.

As the parameters of the backbone nnU-Net, DGC-Encoder, and DGC-Decoder are jointly learned in DGC-Seg model, the combined loss function L_{com} is defined as

$$L_{com} = L_{seg} + \lambda L_{dgc} \tag{14}$$

where λ is a parameter to balance the segmentation loss and the graph-based reasoning loss, and λ is empirically set as 0.2.

4. Experiments

4.1. Datasets and implementation details

2019 Kidney Tumor segmentation challenge dataset [39] is used to evaluate the performance of the proposed DGC-Seg model. There are 210 patient CT scans and corresponding manual segmentations (referred to as ground truth (GT)). The voxel size of CT volumes varies because the data is collected from multiple hospitals or different scanners. We resampled all the CT volumes using voxel size of $1.99 \times 1.99 \times 1.99\ mm^3$. Data augmentation was performed, including random scaling, random rotation, horizontal and vertical mirrors, brightness, and gamma noise augmentations. 20% of the 210 cases are randomly selected as the testing set. For the remaining 168 cases, we randomly separated them into five sub-sets where four sets are used for training and one set for validation.

Our model was implemented by PyTorch framework on a single NVIDIA RTX 2080Ti (11GB RAM). The patch size is $128 \times 128 \times 128$, and batch size is 2. Adam is used as the optimizer. The initial learning rate is 0.01, followed by a poly learning policy where the initial learning rate is multiplied by $(1 - \frac{iter}{total\_iter})^{0.9}$ for decay. The detailed parameters of the backbone network and DGC autoencoder are given in Table 1.

4.2. Evaluation metrics

The segmentation performance is evaluated by Dice coefficient (DSC) [40] and intersection over union (IoU) [41] in terms of spatial volumetric overlapping, and Hausdorff distance (HD) [42] with respect to shape similarity.

Dice of tumor is defined as

$$Dice_{tumor} = \frac{2|P_{tumor} \cap L_{tumor}|}{|P_{tumor}| + |L_{tumor}|} \tag{15}$$

where P_{tumor} and L_{tumor} represent the segmentation result and GT. $Dice_{tumor}$ is within the range of 0 and 1, and a greater value indicates better segmentation result. Similarly, we can obtain the Dice of kidney as $Dice_{kidney}$. The IoU of tumor is obtained as

$$IoU_{tumor} = \frac{|P_{tumor} \cap L_{tumor}|}{|P_{tumor} \cup L_{tumor}|} \tag{16}$$

Similarly, we can also get the IoU of kidney as IoU_{kidney}.

The HD between the segmented tumor boundary and GT boundary is defined as

$$HD_{tumor}(P_{tumor}, L_{tumor}) = \max\{h(P_{tumor}, L_{tumor}), h(L_{tumor}, P_{tumor})\} \tag{17}$$

where $h(P_{tumor}, L_{tumor})$ denotes the distance between the surfaces of P_{tumor} and L_{tumor},

$$h(P_{tumor}, L_{tumor}) = \max_{a \in P_{tumor}} \min_{b \in L_{tumor}} \|a - b\| \tag{18}$$

where a and b are from P_{tumor} and L_{tumor} respectively. Smaller HD_{tumor} value indicates better segmentation results. Similarly, we can calculate the HD of kidney.

Table 1
Parameter settings of layers in the backbone and the proposed DGC autoencoder. In this table, k denotes Conv3d kernel, C denotes channels, p denotes padding, s denotes stride, ins_norm denotes Instance Normalization.

Backbone (3D nnU-Net)					
Encoder	Operations	Output size	Decoder	Operations	Output size
Input		1 × 128^3	Input		320 × 4^3
Encoder_1	Conv3d, k=3 × 3 × 3, C=32, p=1, s=1; ins_norm. Conv3d, k=3×3 × 3, C=32, p=1, s=1; ins_norm.	32 × 128^3	Decoder_1	TransposeConv3d, k=2 × 2 × 2, C=320, s=2. Conv3d, k=3 × 3 × 3, C=320, p=1, s=1; ins_norm. Conv3d, k=3 × 3 × 3, C=320, p=1, s=1; ins_norm.	320 × 8^3
Encoder_2	Conv3d, k=3 × 3 × 3, C=64, p=1, s=1; ins_norm. Conv3d, k=3×3 × 3, C=64, p=1, s=2; ins_norm.	64 × 64^3	Decoder_2	TransposeConv3d, k=2 × 2 × 2, C=256, s=2. Conv3d, k=3 × 3 × 3, C=256, p=1, s=1; ins_norm. Conv3d, k=3 × 3 × 3, C=256, p=1, s=1; ins_norm.	256 × 16^3
Encoder_3	Conv3d, k=3 × 3 × 3, C=128, p=1, s=1; ins_norm. Conv3d, k=3 × 3 × 3, C=128, p=1, s=2; ins_norm.	128 × 32^3	Decoder_3	TransposeConv3d, k=2 × 2 × 2, C=128, s=2. Conv3d, k=3 × 3 × 3, C=128, p=1, s=1; ins_norm. Conv3d, k=3 × 3 × 3, C=128, p=1, s=1; ins_norm.	128 × 32^3
Encoder_4	Conv3d, k=3 × 3 × 3, C=256, p=1, s=1; ins_norm. Conv3d, k=3×3 × 3, C=256, p=1, s=2; ins_norm.	256 × 16^3	Decoder_4	TransposeConv3d, k=2 × 2 × 2, C=64, s=2. Conv3d, k=3 × 3 × 3, C=64, p=1, s=1; ins_norm. Conv3d, k=3 × 3 × 3, C=64, p=1, s=1; ins_norm.	64 × 64^3
Encoder_5	Conv3d, k=3 × 3 × 3, C=320, p=1, s=1; ins_norm. Conv3d, k=3×3 × 3, C=320, p=1, s=2; ins_norm.	320 × 8^3	Decoder_5	TransposeConv3d, k=2 × 2 × 2, C=32, s=2. Conv3d, k=3 × 3 × 3, C=32, p=1, s=1; ins_norm. Conv3d, k=3 × 3 × 3, C=32, p=1, s=1; ins_norm.	32 × 128^3
Encoder_6	Conv3d, k=3 × 3 × 3, C=320, p=1, s=1; ins_norm. Conv3d, k=3×3 × 3, C=320, p=1, s=2; ins_norm.	320 × 4^3	Decoder_6	Conv3d, k=1 × 1 × 1, C=3. softmax	3 × 128^3
DGC autoencoder					
Encoder	Operations	Output size	Decoder	Operations	Output size
Input		64 × 320	Input		64 × 80
Encoder_1	$\mathbf{Y}_1 = f_{LeakyReLU}(\mathbf{A}_1\widetilde{\mathbf{X}}\mathbf{W}_1)$	64 × 160	Decoder_1	$\widehat{\mathbf{Y}}_1 = f_{LeakyReLU}(\mathbf{A}_1\mathbf{Y}_2\widehat{\mathbf{W}}_1)$	64 × 160
Encoder_2	$\mathbf{Y}_2 = f_{LeakyReLU}(\widetilde{\mathbf{A}}_2\mathbf{Y}_1\mathbf{W}_2)$	64 × 80	Decoder_2	$\widehat{\mathbf{Y}}_2 = f_{LeakyReLU}(\mathbf{A}_1\widehat{\mathbf{Y}}_1\widehat{\mathbf{W}}_2)$	64 × 320

4.3. Ablation studies

Ablation studies are performed to evaluate the contributions of each of the major components in the proposed DGC-Seg model. The experimental results are given in Table 2. The baseline 3D nnU-Net model achieved $Dice_{kidney}$ of 0.9601, IoU_{kidney} of 0.9237 and HD_{kidney} of 17.8982 mm. The baseline tumor segmentation results are $Dice_{tumor}$ of 0.8223, IoU_{tumor} of 0.7330, and HD_{tumor} of 34.3687 mm. By considering graph convolutional autoencoder without dynamic strategy (referred to as GCN), $Dice_{tumor}$ was improved by 3.08%, and IoU_{tumor} increased by 2.4%. The segmentation shape similarity, however, was decreased by 2.58 mm in terms of HD_{kidney} and 3.32 mm with respect to HD_{tumor}. With DGC, HD_{kidney} and HD_{tumor} results were significantly improved by 2.83 mm and 7.29 mm. With NodeAttri-Attention, the model reached the best $Dice_{kidney}$ of 96.13%, IoU_{kidney} of 92.59%, HD_{kidney} of 17.57 mm. The results on tumor segmentation was significantly better than the baseline model with 4.26% higher $Dice_{tumor}$, 3.93% greater IoU_{tumor} and 5.29% better HD_{tumor}.

Nine examples of kidney and tumor segmentation results are given in Fig. 4. Overall, the segmented tumors by nnU-Net were relatively smaller than the GT. For cases in Fig. 4(b), (f) and (h), the backbone network failed to localize the target tumor when the region is small and has similar pattern and texture features as kidney region. Graph network enhanced the relation inference and reasoning and detected tumor in those three cases. Dynamic evolution of graph by DGC further improved the complete tumor definition and NodeAttri-Attention contributed to fine-detailed boundary identification. For the large tumor case in Fig. 4(c), there are leakages to kidney region by nnU-Net and GCN. DGC dynamically updated graph connection and topology during the training process, which improved the boundary definition especially when there are weak boundaries between different objects such as tumor and kidney. In conclusion, dynamic graph evolution and NodeAttri-attention improved the complete tumor identification and weak boundary definition especially when the tumor is small, has similar texture features as the kidney region or weak boundaries.

4.4. Investigation of different segmentation backbones

To demonstrate the generality of the proposed DGC autoencoder with NodeAttri-Attention, we performed further experiments by using different segmentation backbones, including 3D U-Net [43], 3D nnU-Net [36], and 3D ResNet [44]. The results are given in Table 3.

As shown, our model consistently improved the backbone segmentation results on both kidney and tumor. The worst results were observed when using 3D U-Net as the backbone. Dice and IoU were slightly improved when embedding NodeAttri-Attention DGC autoencoder to 3D U-Net and 3D nnU-Net, while

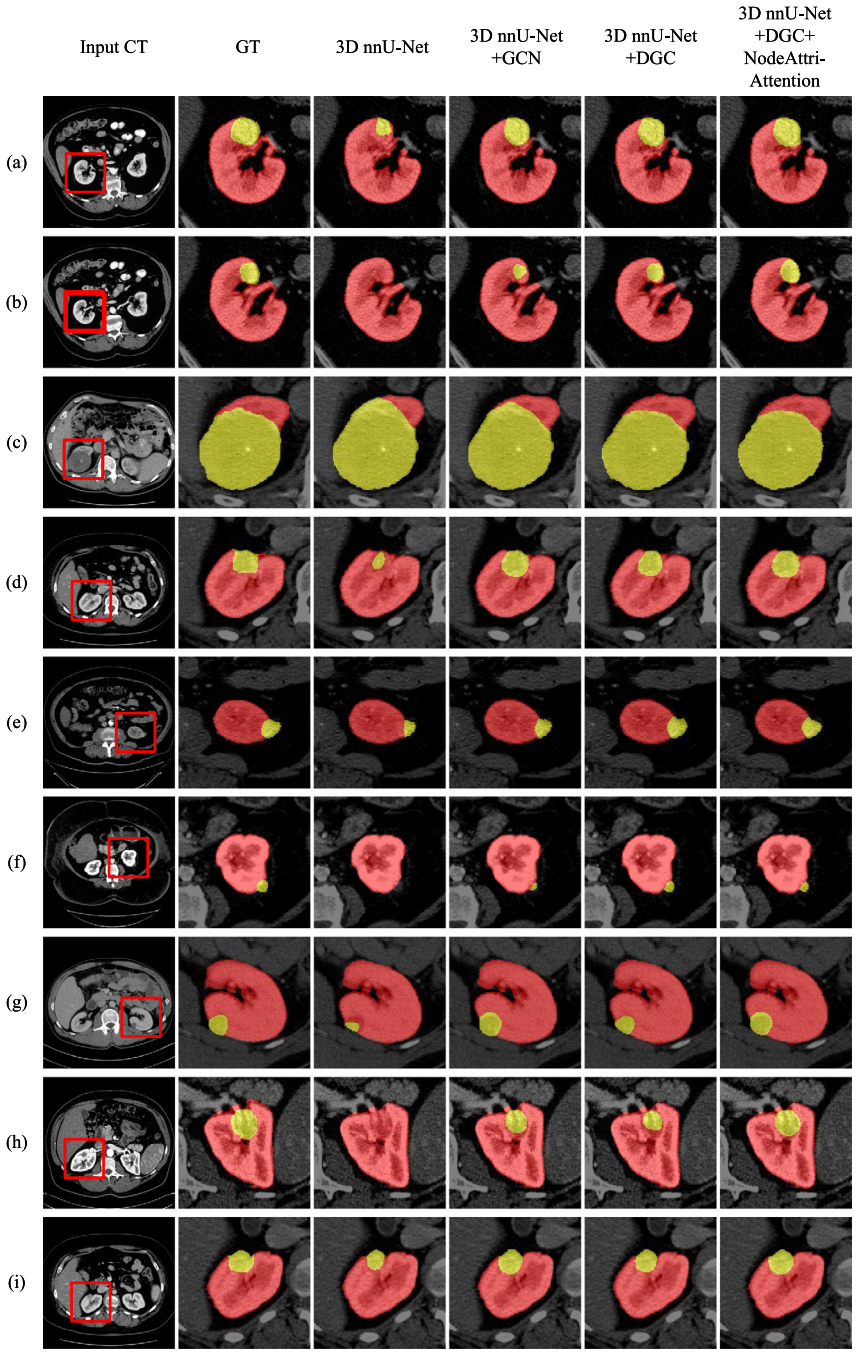

Fig. 4. Kidney and tumor segmentation results by baseline nnU-Net, nnU-Net with graph convolutional autoencoder (nnU-Net+GCN), nnU-Net with dynamic graph convolutional autoencoder (nnU-Net+DGC), and our final model with NodeAttri-Attention DGC. Dynamic graph evolution and NodeAttri-attention improved the complete tumor identification and weak boundary definition especially when the tumor is small or has similar texture features as kidney region. Segmented kidney and tumor are shown by red and yellow colormaps.

Table 2
Ablation study results of node-attribute-wise attention (NodeAttri-Attention), graph convolutional autoencoder (GCN), and dynamic graph convolutional autoencoder (DGC)

NodeAttri-Attention	GCN	DGC	$Dice_{kidney}$	IoU_{kidney}	HD_{kidney}(mm)	$Dice_{tumor}$	IoU_{tumor}	HD_{tumor}(mm)
×	×	×	0.9601	0.9237	17.8982	0.8223	0.7330	34.3687
×	✓	×	0.9603	0.9242	20.4805	0.8531	0.7570	37.6906
×	×	✓	0.9597	0.9230	17.6508	0.8585	0.7641	**30.4047**
✓	×	✓	**0.9613**	**0.9259**	**17.5710**	**0.8649**	**0.7722**	33.8393

HD results were significantly improved. Particularly, NodeAttri-Attention DGC autoencoder increased HD_{kidney} by 8.23 mm and HD_{tumor} by 9.23 mm when embedded with 3D U-Net. The best results were achieved when using 3D ResNet as the basic segmentation framework. This is especially the case for HD where HD_{tumor} and HD_{kidney} were improved by 28.50 mm and 3.04 mm.

Table 3
Segmentation results of embedding NodeAttri-Attention DGC autoencoder (ours) to different segmentation backbones.

	$Dice_{kidney}$	IoU_{kidney}	HD_{kidney}(mm)	$Dice_{tumor}$	IoU_{tumor}	HD_{tumor}(mm)
3D U-Net	0.9592	0.9220	27.1093	0.8176	0.7213	43.4326
3D U-Net+ours	0.9595	0.9228	18.8745	0.8397	0.7459	34.2011
3D nnU-Net	0.9601	0.9237	17.8982	0.8223	0.7330	34.3687
3D nnU-Net+ours	0.9613	0.9259	17.5710	**0.8649**	**0.7722**	33.8393
3D ResNet	0.9589	0.9216	19.4117	0.8314	0.7404	52.0586
3D ResNet+ours	**0.9615**	**0.9262**	**16.3665**	0.8592	0.7669	**23.5577**

Fig. 5. Segmentation results of embedding NodeAttri-Attention DGC autoencoder (ours) with different segmentation backbones. The segmented kidney and tumor regions are shown by red and yellow colormaps.

Nine examples of segmentation results are given in Fig. 5. Overall, NodeAttri-Attention DGC improved the shape similarity when compared with different backbones. Our first finding is that when there are weak boundaries between tumor and kidney such as Fig. 5(b), (c), (f) and (i), 3D U-Net and 3D nnU-Net failed to identify the boundaries.

NodeAttri-Attention DGC autoencoder successfully defined the boundaries. 3D ResNet results were better than the other two backbones, and NodeAttri-Attention DGC autoencoder further improved the weak boundary definitions. Secondly, when the tumor has similar appearance and textures to kidney regions, 3D U-Net and 3D nnU-Net did not detect the tumor regions, especially when the tumor is small, as shown in Fig. 5(a) and (e). NodeAttri-Attention DGC autoencoder assisted the information propagation

Table 4
Model efficiency when embedding NodeAttri-Attention DGC auto-encoder (ours) to different segmentation backbones.

	3D U-Net	3D U-Net+ours	3D nnU-Net	3D nnU-Net+ours	3D ResNet	3D ResNet+ours
Average training time (seconds per epoch)	374.42	376.13	392.68	396.00	494.81	497.81
Average testing time (seconds per patient)	29.57	29.67	34.18	34.37	39.49	39.70

Table 5
Comparison of segmentation results with other state-of-the-art methods.

Method	$Dice_{kidney}$	IoU_{kidney}	HD_{kidney}(mm)	$Dice_{tumor}$	IoU_{tumor}	HD_{tumor}(mm)
2D_PSPNET [10]	0.902	–	–	0.638	–	–
3D_FCN_PPM [35]	0.927	–	–	0.802	–	–
3D U-Net [43]	0.959	0.922	27.109	0.818	0.721	43.433
3D nnU-Net [36]	0.960	0.924	17.898	0.822	0.733	34.369
MSS U-net [8]	0.958	0.920	21.123	0.821	0.720	49.347
nnU-Net_with_graph_reasoning [21]	0.938	0.890	20.131	0.853	0.756	36.574
ours (3D nnU-Net backbone)	**0.961**	**0.926**	**17.571**	**0.865**	**0.772**	**33.839**

Table 6
Comparison with other convolution layers using 3D nnU-Net backbone.

	$Dice_{kidney}$	IoU_{kidney}	HD_{kidney}(mm)	$Dice_{tumor}$	IoU_{tumor}	HD_{tumor}(mm)
3D nnU-Net	0.960	0.924	17.898	0.822	0.733	34.369
+ Ours (DGC by Conv)	0.956	0.916	20.875	0.838	0.736	39.049
+ Ours (DGC by Dilated-Conv)	0.958	0.921	18.903	0.844	0.740	37.760
+ Ours	0.961	0.926	17.571	0.865	0.772	33.839

across regions and detected the tumor located inside the kidney successfully. The two findings can be explained by the dynamic information evolution and propagation by DGC during the training process, which improved the localization and boundary definition of objects with weak boundaries, irregular shapes and various sizes.

We further analyze the model efficiency when using the newly proposed NodeAttri-Attention DGC component with different backbones. We report the average time taken in each epoch during the training process, and the average time segmenting a whole patient CT scan using the trained model. The results are given in Table 4. It is noted that during the training process, one volumetric patch of 128 × 128 × 128 is randomly cropped from a patient's scan, resulting in 135 patches in one epoch. When using a trained model to segment a whole CT scan, volumetric patches are extracted using a sliding window. The number of patches that can be extracted from a patient varies, where there are averagely 10.5 volumetric patches per patient in the testing dataset. As shown by the result, our NodeAttri-Attention DGC increased the training time of 3D U-Net, 3D nnU-Net, and 3D ResNet by 1.71, 3.32 and 3 s per epoch. When segmenting one patient using a trained model, the models with NodeAttri-Attention DGC were 0.10 s, 0.19 s, 0.21 s longer than the corresponding backbone models of 3D U-Net, 3D nnU-Net, and 3D ResNet. The results show that using our newly proposed modules improved the segmentation accuracy. Meanwhile, there are no major changes in training and testing efficiency.

4.5. Comparison with other methods

To further evaluate the performance of DSC-Seg model on kidney and tumor segmentation, we compare with state-of-the-art kidney and tumor segmentation methods including (1) 2D_PSPNET [10], (2) 3D_FCN_PPM [35], (3) 3D U-net [43], (4) 3D nnU-Net [36], (5) MSS U-net [8] and (6) nnU-Net with graph reasoning [21]. The results are given in Table 5.

As shown, our model achieved the best kidney and tumor segmentation results in terms of spatial overlap and shape similarity. For tumor segmentation, our model achieved the highest Dice of 0.865, which was 22.7% higher than 2D_PSPNET, 6.3% better than 3D_FCN_PPM, 4.7% higher than 3D U-net, 4.3% higher than 3D nnU-Net, 4.4% and 1.2% higher than MSS U-net and nnU-Net with graph reasoning. In terms of IoU_{tumor}, our model achieved the best IoU of 0.772, which was 5.1%, 3.9%, 5.2%, 1.6% higher than 3D U-net, 3D nnU-Net, MSS U-net, and nnU-Net with graph reasoning. Our model also yielded the best HD of 33.839 mm, which was 9.58 mm, 0.53 mm, 15.51 mm, 2.73 mm better than other methods, respectively. Even though graph reasoning [21] achieved good performance on tumors, the evaluation results on kidney were relatively low.

Our first finding is that the kidney tumor segmentation task is more challenging than kidney segmentation. This may be caused by two reasons. Firstly, as shown in Figs. 4 and 5, kidney tumors are of diverse sizes, shapes, and locations, whereas kidneys are relatively large and of similar appearance across different patients. Secondly, tumors can be small and have similar texture features as their surrounding or invaded kidney regions, posting challenges to identify the weak boundaries. The second finding is that graph reasoning [21] by projection and re-projection process in graph construction resulted in information loss, resulting in relatively low kidney segmentation results. In comparison, our proposed graph construction strategy and DGC spread the spatial connections between the local areas of the kidneys and tumor during multiple encoding layers. Compared with static graphs, our graph topology and node attributes dynamically evolve as the encoding layer gradually deepens. The dynamic evolution contributes to the reasoning of semantic information and spatial dependencies between image regions, especially those with long distances. Thus, the segmentation results of kidney and tumor outperformed the conventional graph reasoning model, especially the shape similarity.

4.6. Comparison with other convolution layers

To further demonstrate the effectiveness of DGC in capturing global information, we compare DGC with convolution layers and dilated convolutions using a similar number of parameters.

We first calculate the number of parameters in DGC by torchinfo.[1] Our DGC has 128,000 parameters. Given the similar number

[1] https://github.com/TylerYep/torchinfo

of parameters, we can obtain one convolution layer with kernel size = 3 × 3 × 3, padding = 1, stride = 1, input channels = 320 and output channels 15 (total 129,615 parameters). Dilated convolution operation can be obtained by kernel size = 3 × 3 × 3, stride = 1, input channels = 320, output channels = 8, padding =1 for dilated-rate 1 and padding =2 for dilated-rate 2 (total 138,256 parameters). Afterwards, we replace DGC by convolution layer (referred to as Ours (DGC by Conv)) and dilated convolutions (referred to as Ours (DGC by Dilated-Conv)), respectively. The other parameter settings and implementation details are the same as our original model. As shown by Table 6, our DGC outperformed the other two approaches in terms of all evaluation measures. The results demonstrated the capacity of DGC in extracting and propagating global information when using similar number of parameters.

5. Conclusion

We propose a NodeAttri-Attention enhanced DGC autoencoder to extract and integrate semantic connections, spatial relations, and dependencies in volumetric image segmentation, especially when there are long-distance objects with close semantic relations and neighboring objects with indistinct boundaries. NodeAttri-Attention mechanism obtains attention-enhanced node attributes by discriminating adaptive contributions of various features. Node connections represent topological relations and high-level correlations. Finally, DGC strategy dynamically updates the graph to learn and integrate the complex and underlying correlations across image regions. The effectiveness of DGC autoencoder and NodeAttri-Attention are validated by kidney and tumor segmentation from CT and several biomedical image segmentation backbones. Experimental results and ablation studies demonstrate the improved performance and effectiveness of each of our major innovations, especially for tumors.

CRediT authorship contribution statement

Ping Xuan: Designed the method and participated in manuscript writing. **Hui Cui:** Participated in method design and manuscript writing. **Hongda Zhang:** Designed the experiments and edited the manuscript. **Tiangang Zhang:** Participated in manuscript writing. **Linlin Wang:** Participated in experiment design. **Toshiya Nakaguchi:** Participated in experiment design. **Henry B.L. Duh:** Participated in manuscript writing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work is supported by the Natural Science Foundation of China (61972135), the Natural Science Foundation of Heilongjiang Province (LH2019F049), the China Postdoctoral Science Foundation (2019M650069), and the Fundamental Research Foundation of Universities in Heilongjiang Province for Technology Innovation (KJCX201805).

All authors contributed to the article and approved the submitted version.

References

[1] G. Litjens, T. Kooi, B.E. Bejnordi, A.A.A. Setio, F. Ciompi, M. Ghafoorian, J.A. Van Der Laak, B. Van Ginneken, C. Sánchez, A survey on deep learning in medical image analysis, Med. Image Anal. 42 (2017) 60–88.
[2] R. Cuingnet, R. Prevost, D. Lesage, L.D. Cohen, B. Mory, R. Ardon, Automatic detection and segmentation of kidneys in 3D CT images using random forests, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2012, pp. 66–74.
[3] N. Heller, F. Isensee, K.H. Maier-Hein, X. Hou, C. Xie, F. Li, Y. Nan, G. Mu, Z. Lin, M. Han, The state of the art in kidney and kidney tumor segmentation in contrast-enhanced ct imaging: Results of the kits19 challenge, Med. Image Anal. 67 (2019) 101821.
[4] Q. Jin, H. Cui, C. Sun, Z. Meng, R. Su, Cascade knowledge diffusion network for skin lesion diagnosis and segmentation, Appl. Soft Comput. (2020) 106881.
[5] H. Cui, Y. Xu, W. Li, L. Wang, H. Duh, Collaborative learning of cross-channel clinical attention for radiotherapy-related esophageal fistula prediction from CT, in: A.L. Martel, P. Abolmaesumi, D. Stoyanov, D. Mateus, M.A. Zuluaga, S.K. Zhou, D. Racoceanu, L. Joskowicz (Eds.), International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2020, pp. 212–220.
[6] Y. Guo, Y. Liu, T. Georgiou, M.S. Lew, A review of semantic segmentation using deep neural networks, Int. J. Multimed. Inf. Retr. 7 (2) (2018) 87–93.
[7] D.M. Pelt, J.A. Sethian, A mixed-scale dense convolutional neural network for image analysis, Proc. Natl. Acad. Sci. 115 (2) (2018) 254–259.
[8] W. Zhao, D. Jiang, J. Peña Queralta, T. Westerlund, MSS U-Net: 3D segmentation of kidneys and tumors from CT images with a multi-scale supervised U-Net, Inform. Med. Unlocked 19 (2020).
[9] Z. Li, J. Pan, H. Wu, Z. Wen, J. Qin, Memory-efficient automatic kidney and tumor segmentation based on non-local context guided 3D U-Net, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2020, pp. 197–206.
[10] H. Zhao, J. Shi, X. Qi, X. Wang, J. Jia, Pyramid scene parsing network, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2881–2890.
[11] H. Zhao, Y. Zhang, S. Liu, J. Shi, C. Change Loy, D. Lin, J. Jia, Psanet: Point-wise spatial attention network for scene parsing, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 267–283.
[12] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, Y. Wei, Deformable convolutional networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 764–773.
[13] X. Wang, A. Gupta, Videos as space-time region graphs, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 399–417.
[14] X. Wang, R. Girshick, A. Gupta, K. He, Non-local Neural Networks, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 7794–7803.
[15] Y. Chen, Y. Kalantidis, J. Li, S. Yan, J. Feng, A^2-Nets: Double attention networks, 2018, CoRR, abs/1810.11579.
[16] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, H. Lu, Dual attention network for scene segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 3146–3154.
[17] H. Fan, P. Chu, L.J. Latecki, H. Ling, Scene parsing via dense recurrent neural networks with attentional selection, in: 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), IEEE, 2019, pp. 1816–1825.
[18] B. Shuai, Z. Zuo, B. Wang, G. Wang, Scene segmentation with dag-recurrent neural networks, IEEE Trans. Pattern Anal. Mach. Intell. 40 (6) (2017) 1480–1493.
[19] Y. Li, A. Gupta, Beyond grids: Learning graph representations for visual recognition, in: Advances in Neural Information Processing Systems, 2018, pp. 9225–9235.
[20] X. Liang, Z. Hu, H. Zhang, L. Lin, E.P. Xing, Symbolic graph reasoning meets convolutions, in: Advances in Neural Information Processing Systems, 2018, pp. 1853–1863.
[21] Y. Chen, M. Rohrbach, Z. Yan, Y. Shuicheng, J. Feng, Y. Kalantidis, Graph-based global reasoning networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 433–442.
[22] S. Mo, M. Cai, L. Lin, R. Tong, Q. Chen, F. Wang, H. Hu, Y. Iwamoto, X.-H. Han, Y.-W. Chen, Multimodal priors guided segmentation of liver lesions in MRI using mutual information based graph co-attention networks, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2020, pp. 429–438.
[23] T.N. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, 2016, ArXiv Preprint ArXiv:.02907.
[24] X. Li, Y. Yang, Q. Zhao, T. Shen, Z. Lin, H. Liu, Spatial Pyramid Based Graph Reasoning for Semantic Segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2020, pp. 8950–8959.
[25] J. Hu, L. Shen, G. Sun, Squeeze-and-excitation networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 7132–7141.

[26] T.N. Kipf, M.J.a.p.a. Welling, Semi-supervised classification with graph convolutional networks, 2016, ArXiv Preprint ArXiv:.02907.
[27] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, S.Y. Philip, A comprehensive survey on graph neural networks, IEEE Trans. Neural Netw. Learn. Syst. (2020).
[28] P. Xuan, S. Pan, T. Zhang, Y. Liu, H.J.C. Sun, Graph convolutional network and convolutional neural network based method for predicting lncRNA-disease associations, Cells 8 (9) (2019) 1012.
[29] P. Xuan, L. Gao, N. Sheng, T. Zhang, T. Nakaguchi, Graph convolutional autoencoder and fully-connected autoencoder with attention mechanism based method for predicting drug-disease associations, IEEE J. Biomed. Health Inform. (2020).
[30] Z. Ying, J. You, C. Morris, X. Ren, W. Hamilton, J. Leskovec, Hierarchical graph representation learning with differentiable pooling, in: Advances in Neural Information Processing Systems, 2018, pp. 4800–4810.
[31] G. Bertasius, L. Torresani, S.X. Yu, J. Shi, Convolutional random walk networks for semantic image segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 858–866.
[32] G. Bertasius, L. Torresani, S.X. Yu, J. Shi, Convolutional random walk networks for semantic image segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 858–866.
[33] H. Cui, X. Wang, J. Zhou, M. Fulham, S. Eberl, D. Feng, Topology constraint graph-based model for non-small-cell lung tumor segmentation from PET volumes, in: 2014 IEEE 11th International Symposium on Biomedical Imaging (ISBI), 2014, pp. 1243–1246.
[34] H. Cui, X. Wang, J. Zhou, G. Gong, S. Eberl, Y. Yin, L. Wang, D. Feng, M. Fulham, A topo-graph model for indistinct target boundary definition from anatomical images, Comput. Methods Programs Biomed. 159 (2018) 211–222.
[35] G. Yang, G. Li, T. Pan, Y. Kong, J. Wu, H. Shu, L. Luo, J.-L. Dillenseger, J.-L. Coatrieux, L. Tang, Automatic segmentation of kidney and renal tumor in ct images based on 3d fully convolutional neural network with pyramid pooling module, in: 2018 24th International Conference on Pattern Recognition (ICPR), IEEE, 2018, pp. 3790–3795.
[36] F. Isensee, P.F. Jaeger, S.A. Kohl, J. Petersen, K.H. Maier-Hein, NnU-Net: a self-configuring method for deep learning-based biomedical image segmentation, Nature Methods (2020) 1–9.
[37] F. Isensee, J. Petersen, A. Klein, D. Zimmerer, P.F. Jaeger, S. Kohl, J. Wasserthal, G. Koehler, T. Norajitra, S. Wirkert, et al., NnU-Net: Self-adapting framework for U-net-based medical image segmentation, 2018, arXiv preprint arXiv:1809.10486.
[38] C. Chen, Q. Dou, Y. Jin, H. Chen, J. Qin, P.-A. Heng, Robust multimodal brain tumor segmentation via feature disentanglement and gated fusion, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2019, pp. 447–456.
[39] N. Heller, N. Sathianathen, A. Kalapara, E. Walczak, K. Moore, H. Kaluzniak, J. Rosenberg, P. Blake, Z. Rengel, M. Oestreich, The kits19 challenge data: 300 kidney tumor cases with clinical context, ct semantic segmentations, and surgical outcomes, 2019, ArXiv Preprint ArXiv:.00445.
[40] K.H. Zou, S.K. Warfield, A. Bharatha, C.M. Tempany, M.R. Kaus, S.J. Haker, W.M. Wells III, F.A. Jolesz, R. Kikinis, Statistical validation of image segmentation quality based on a spatial overlap index1: scientific reports, Academic Radiol. 11 (2) (2004) 178–189.
[41] G. Csurka, D. Larlus, F. Perronnin, F. Meylan, What is a good evaluation measure for semantic segmentation?, in: BMVC, Vol. 27, p. 2013.
[42] D.P. Huttenlocher, G.A. Klanderman, W.J. Rucklidge, Comparing images using the Hausdorff distance, IEEE Trans. Pattern Anal. Mach. Intell. 15 (9) (1993) 850–863.
[43] O. Çiçek, A. Abdulkadir, S.S. Lienkamp, T. Brox, O. Ronneberger, 3D U-net: learning dense volumetric segmentation from sparse annotation, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2016, pp. 424–432.
[44] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.