Structure Embedded Nucleus Classification for Histopathology Images

Wei Lou, Xiang Wan, Guanbin Li, *Member, IEEE*, Xiaoying Lou, Chenghang Li, Feng Gao, and Haofeng Li, *Member, IEEE*

***Abstract*— Nuclei classification provides valuable information for histopathology image analysis. However, the large variations in the appearance of different nuclei types cause difficulties in identifying nuclei. Most neural network based methods are affected by the local receptive field of convolutions, and pay less attention to the spatial distribution of nuclei or the irregular contour shape of a nucleus. In this paper, we first propose a novel polygon-structure feature learning mechanism that transforms a nucleus contour into a sequence of points sampled in order, and employ a recurrent neural network that aggregates the sequential change in distance between key points to obtain learnable shape features. Next, we convert a histopathology image into a graph structure with nuclei as nodes, and build a graph neural network to embed the spatial distribution of nuclei into their representations. To capture the correlations between the categories of nuclei and their surrounding tissue patterns, we further introduce edge features that are defined as the background textures between adjacent nuclei. Lastly, we integrate both polygon and graph structure learning mechanisms into a whole framework that can extract intra and inter-nucleus structural characteristics for nuclei classification. Experimental results show that the proposed framework achieves significant improvements compared to the previous methods. Code and data are made available via https://github.com/lhaof/SENC**

***Index Terms*— Nuclei classification, recurrent neural network, graph neural network.**

Manuscript received 6 November 2023; revised 1 March 2024; accepted 6 April 2024. Date of publication 12 April 2024; date of current version 3 September 2024. This work was supported in part by Chinese Key-Area Research and Development Program of Guangdong Province under Grant 2020B0101350001; in part by Guangdong Basic and Applied Basic Research Foundation under Grant 2023A1515011464; in part by the National Natural Science Foundation of China under Grant 62102267; in part by Guangdong Provincial Key Laboratory of Big Data Computing, The Chinese University of Hong Kong, Shenzhen; and in part by the National Key Clinical Discipline. *(Corresponding author: Haofeng Li.)*

This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by The Sixth Affiliated Hospital, Sun Yat-sen University.

Wei Lou and Haofeng Li are with Shenzhen Research Institute of Big Data, Guangdong Provincial Key Laboratory of Big Data Computing, The Chinese University of Hong Kong at Shenzhen, Shenzhen 518172, China (e-mail: weilou@link.cuhk.edu.cn; lhaof@sribd.cn).

Xiang Wan is with Shenzhen Research Institute of Big Data, Guangdong Provincial Key Laboratory of Big Data Computing, The Chinese University of Hong Kong at Shenzhen, Shenzhen 518172, China, and also with the Pazhou Laboratory, Guangzhou 510330, China (e-mail: wanxiang@sribd.cn).

Guanbin Li is with the School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou 510006, China (e-mail: liguanbin@mail.sysu.edu.cn).

Xiaoying Lou is with the Department of Pathology, Guangdong Provincial Key Laboratory of Colorectal and Pelvic Floor Diseases, The Sixth Affiliated Hospital, Sun Yat-sen University, Guangzhou 510655, China (e-mail: louxy3@mail.sysu.edu.cn).

Chenghang Li is with the Artificial Intelligence Thrust, The Hong Kong University of Science and Technology at Guangzhou, Guangzhou 510030, China (e-mail: cli136@connect.hkust-gz.edu.cn).

Feng Gao is with the Department of Colorectal Surgery, Department of General Surgery, Guangdong Provincial Key Laboratory of Colorectal and Pelvic Floor Diseases, The Sixth Affiliated Hospital, Sun Yat-sen University, Guangzhou 510655, China, and also with Shanghai Artificial Intelligence Laboratory, Shanghai 200240, China (e-mail: gaof57@mail.sysu.edu.cn).

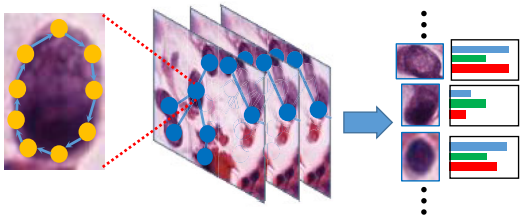
Digital Object Identifier 10.1109/TMI.2024.3388328

I. Introduction

RECENTLY, computer-aided diagnosis (CAD) systems have achieved great success in histological examination tasks for their efficient, accurate and reproducible diagnosis performance [1], [2]. In a CAD system, nuclei segmentation and classification are critical steps for analyzing histopathology images. Segmenting a nucleus is to label all its pixels while nuclei classification is to identify the category of a nucleus. Solving these tasks not only obtains the size, texture and shape of each nucleus, but also provides the distribution among different nuclei types. The spatial relationships among different types of nuclei can effectively guide the CAD system in computational pathology tasks such as survival prediction [3], [4], cancer subtype classification and grading [5], [6].

Nowadays, nuclei classification for histopathology images remains a challenge. Two nuclei of different types could have similar shapes and textures, and it is hard to distinguish between them. Meanwhile, for the nuclei of the same category, their appearances could have a wide variation during the different periods of their life cycle [7], [8]. It is difficult for CAD systems or inexperienced pathologists to classify every single nucleus in an image accurately. Thus, we aim at solving the nuclei classification task.

Most deep learning (DL) based methods [7], [8], [9], [10] adopt Convolutional Neural Networks (CNNs) to detect and classify a nucleus based on its own convolutional features. However, these approaches seldom consider the relationship between different nuclei in an image. Such inter-nucleus information is crucial and widely used by experts in manual classification. Moreover, existing DL-based models rely on



(a) Polygon Structure (b) Graph Structure (c) Nuclei Classification

Fig. 1. Overview of the proposed nucleus classification framework that consists of (a) a polygon structure learning module, (b) a multi-layer GNN for graph structure learning, and (c) an instance-level classifier.

pixel-level features that are implicitly encoded by learnable convolutions. These approaches may be able to capture corner or boundary feature, but they can be distracted by the texture, color information, or overlapping areas of nuclei. Most of them do not disentangle the shape of a nucleus from other attributes or explicitly model it as an individual structure (such as a polygon or a sequence of points). Recently, Graph Neural Networks (GNNs) based methods have grown rapidly in computational pathology [11], [12], [13], [14], [15]. Previous GNN models employ tissue patches or nuclei objects as graph nodes to construct a graph. However, most of them focus on classifying Whole Slide Images (WSIs) instead of nuclei.

Inspired by the above observations, we propose to improve the nuclei classification by considering not only the correlations among different nuclei, but also the polygon structure of a nucleus shape. First, to exploit the shape characteristics of a nucleus, we model a nucleus contour as a polygon, which can be described as an ordered sequence of points sampled from the contour (as shown in Fig. 1(a)). A recurrent neural network (RNN) is utilized to learn structural representations for the polygon from the point sequence, by aggregating the continuous changes in relative positions of sampled contour points. Such a polygon feature can well describe the irregular contour of a nucleus, and will act as a part of the nucleus representation. Second, to harvest the spatial distribution among nuclei, we model a histopathology image as a graph by defining a node as a nucleus and an edge as the middle point of the line connecting two adjacent nuclei. Due to the receptive field of CNNs, an edge feature is supposed to describe a local tissue region centered at the middle point. A graph neural network is exploited to capture inter-nuclei contexts via iteratively propagating each node information to its neighbors and updating each nucleus feature with its background and adjacent nodes. The structural information of the whole histopathology graph is embedded into the enhanced representation of each nucleus. Lastly, we combine the above ideas of graph structure learning and polygon shape learning to develop a nuclei classification framework in which the pixel-level feature extraction and the nuclei-level classification are end-to-end trained.

Our main contributions are as follows:

- A novel intra-nucleus polygon structure learning (PSL) module that learns the shape feature of a nucleus.
- A novel structure embedded nuclei classification framework based on an inter-nucleus graph structure learning (GSL) module and the proposed PSL module.
- The proposed framework significantly outperforms the existing methods by 4.7%-9.8% average F-score on an in-house dataset and three public benchmarks. The experimental results show that both the proposed PSL and GSL modules can effectively improve classification performance.

II. Related Work

A. Cell Classification in Histopathological Images

In the early stage, handcrafted features of texture, morphology and color are extracted and sent into an SVM/AdaBoost classifier for nuclei classification [16], [17]. These methods explicitly model the intra-nucleus structure but are limited by the non-learnable representations. Nowadays, most nuclei classifiers adopt CNNs with two stages, detecting nuclei instances and then labeling them [7], [18], [19], [20], [21]. Sirinukunwattana et al. [22] propose a CNN to detect nuclei centers and another CNN to classify the image patches containing a nucleus. Graham et al. [9] proposes a CNN of three branches, two for segmentation, and one for classification. Doan et al. [8] predicts a weight map to highlight hard pixel samples for classification. However, these approaches are limited by the receptive field of CNNs, and fail to harvest long-range contexts and spatial distributions of nuclei instances. Some non-CNN methods utilize denoised autoencoder [23] or vision transformers [24], [25] to classify cell types. However, they fail to exploit the contour or topology information of cell instances.

B. Cell Nucleus Detection and Segmentation

DL-based methods have achieved remarkable success in nuclei detection and segmentation [26], [27], [28], [29], [30], [31], [32], [33], [34], which are crucial steps as important prerequisites for cell classification. Some approaches employ the two-stage method that first detects the bounding box of each nucleus instance and then proceeds to segment the contour [35], [36], [37]. Some other methods utilize a single-stage framework, which predicts the semantic labels for each pixel and separates each nucleus instance by well-defined distance map or morphology operations [8], [24], [38]. In this work, we use some of the existing nuclei segmentation methods [9], [10], [39], [40] as the pre-processing component of our method and show that the proposed method is flexible to integrate with different nuclei segmentation models to enhance the ability of nuclei classification.

C. Graph Models in Computational Pathology

GNNs have become popular in computational pathology [41], [42], [43], [44], [45], [46], [47]. Most GNN methods are to classify a whole image [48], [49], [50] or tissue patches [51]. In these works, graph nodes are defined as tissue patches [50], [52], nuclei objects [15] or superpixels [14]. The node embeddings can be hand-crafted [41], [53] or extracted from pre-trained models [15], [48]. Different from these works, we design a finer node representation including a shape feature of polygon-structure learning. Some existing

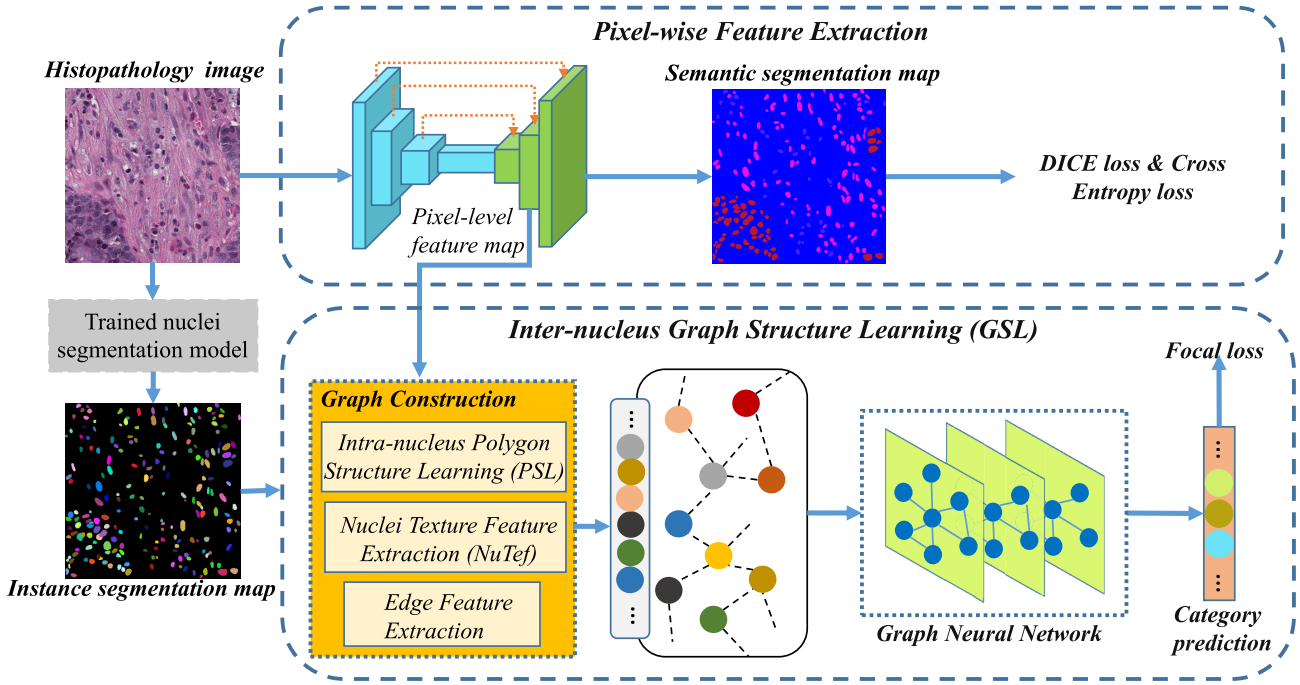


Fig. 2. Structure Embedded Nucleus Classification (SENC) framework. It consists of a pixel-wise feature extraction branch (upper) and an instance-level classification branch (lower) using the inter-nucleus Graph Structure Learning module (GSL). In the GSL module, an intra-nucleus Polygon Structure Learning module (PSL) computes the shape features for nuclei. Then the input image is transformed into a graph and a GNN enhances the features of the graph nodes for nuclei classification.

work is based on GNN but it works by refining the nuclei classification results from existing models [46]. Instead, the GNN in our framework does not rely on nuclei types predicted in advance but directly performs the classification.

D. Contour and Polygon Representation Learning

Predicting polygons for object segmentation [29], [54], [55] and contour detection [56] have been widely studied, while the feature extraction and classification for contours and polygons have been less discussed. Contour-aware nuclei segmentation methods [57] predict if a pixel is at a contour to improve the segmentation, but do not aggregate the pixel-level contour features to classify nuclei. PBC [58] is a polygon-based classifier that pools the texture features of each point in a polygon, but it does not consider the irregularity and smoothness of a polygon contour. Sharma et al. [17] computes morphological statistics (such as Area and Convexity of Contour) as the shape feature of a nucleus. Differently, we model a nucleus contour in a fine-grained way, using a sequence of relative positions between the centroid and vertices.

III. Methodology

A. Structure Embedded Nucleus Classification Framework

The Structure Embedded Nuclei Classification (SENC) framework is illustrated in Fig. 2. The proposed framework consists of a pixel-wise feature extraction branch and an instance-level classification branch based on the module of Inter-nucleus Graph Structure Learning (GSL). The graph construction part in GSL contains a new Polygon Structure Learning (PSL), a Nuclei Texture feature extraction (NuTef) and an edge feature extraction modules.

1) Pixel-Wise Feature Extraction Branch: We utilize an existing CNN [59] as the encoder and a feature pyramid network (FPN) [60] as the decoder to build the pixel-wise feature extraction branch. The branch uses a histopathology image as input and produces a pixel-level feature map from the second last decoder layer. The encoder in the pixel-wise extraction branch consists of four convolutional blocks with large kernel sizes, which help capture a large receptive field for the input image. The output sizes of the four blocks are: $\frac{h}{4} \times \frac{w}{4}$, $\frac{h}{8} \times \frac{w}{8}$, $\frac{h}{16} \times \frac{w}{16}$, $\frac{h}{32} \times \frac{w}{32}$, where h and w are the height and width of the input image, respectively. The decoder consists of three layers. Each decoder output is up-sampled to fuse with the corresponding encoder feature of the same resolution, and the fused result is sent to the next decoder layer. The output of the decoder has the same size of the input image. The encoder-decoder follows a top-down pathway, which involves upsampling the feature maps from higher-level layers to match the resolution of lower-level feature maps, establishing a hierarchy of feature maps with varying levels of spatial information. More detailed values of hyper-parameters can be found in section IV-B. In Fig. 2, the semantic segmentation map is the output of the decoder, when the encoder-decoder is pre-trained on a semantic segmentation task using DICE loss and cross-entropy loss.

The inter-nucleus GSL module takes the pixel-level feature map and an instance segmentation map as inputs. The instance segmentation map is the result of a pre-trained nuclei instance segmentation model. Since we focus on solving the classification task, we simply use an existing nuclei segmentation

model [9] to predict the instance segmentation map. The inter-nucleus GSL module first transforms a histopathology image into a graph structure and then learns nuclei representation as graph nodes and background features as edges. In particular, in the GSL branch, we use the edge feature extraction, the intra-nucleus PSL and the NuTef modules to capture edge features, shape and texture features of nodes, respectively. After enhancing the node features with a graph neural network, all the nuclei are classified with a fully-connected (FC) layer and a Softmax layer.

B. Inter-Nucleus Graph Structure Learning

To capture the spatial relationship among nuclei, we develop an inter-nucleus graph structure learning module, which includes the construction of a histopathology graph topology, and the feature extraction of graph nodes and edges.

1) Histopathology Graph Topology: Suppose a histopathology image contains N nuclei. We construct a graph topology corresponding to the image by defining a graph node as a nucleus. The graph is undirected and can be defined as $G = (V, E)$. V denotes the set of nodes in the graph, namely, all the nuclei entities in the histopathology image. E is the set of edges which represent the inner relationships between adjacent nuclei. To build an efficient and sparsely-connected graph, we first calculate the Euclidean distance between the centroid coordinates of any two nuclei. For each nucleus, K undirected edges are linked between the nucleus and its K nearest neighbors. The connection of nodes can be defined as a symmetric adjacency matrix $A \in R^{N\times N}$, where $A_{u,v} = 1$ if the edge between the u^{th} and v^{th} nodes exists ($e_{u,v} \in E$). After computing the adjacency matrix, the original representations of nodes and edges, we use a GNN to update the node features and to harvest the structural embedded nuclei representation. The updated node features are exploited to predict the nuclei types.

2) Node Feature Extraction: To obtain original node representations, we propose a node feature extraction process. The process takes the feature map from the pixel-wise feature extraction branch (Fig. 2) and a nucleus instance segmentation map as inputs and outputs a feature vector for the nucleus. The shape of the feature map is $\frac{h}{4} \times \frac{w}{4} \times c$. h and w are the height and width of the input image, while c is the number of channels of the feature map. The node feature extraction includes an intra-nucleus Polygon Structure Learning module (PSL) and a Nucleus Texture feature extraction module (NuTef). The PSL module is to calculate the shape feature Z for a nucleus, using a recurrent neural network (RNN). The PSL details are in the next subsection. The NuTef module is to learn the texture feature of a nucleus. As Fig. 4 shows, ROI Align [35] is used with the input feature map to compute the representation $B \in R^{1\times c}$ of the nucleus bounding box. The centroid coordinate (x_0, y_0) of the nucleus is adapted to sample a feature vector $C \in R^{1\times c}$ from the input feature map (of size $\frac{h}{4} \times \frac{w}{4} \times c$). In details, we locate the four pairs of integer coordinates nearest to (x_0, y_0) and sample the centroid feature as the weighted combination of these four feature vectors, using the bilinear interpolation method. To introduce the information of the nucleus location in the original histopathology image, we compute a global position embedding $PE \in R^{1\times c}$ for the nucleus centroid, using the Sinusoidal Position Encoding method [61]. To obtain the final texture representation, the above three feature vectors are concatenated into a vector $T \in R^{1\times 3c}$ as (1) shows:

$$T = concat(\{B, C, P\ E\}). \tag{1}$$

After computing the shape feature Z with the PSL module, the feature vector of a node can be obtained by joining the texture and shape features as: $X = concat\ (T, Z)$.

3) Edge Feature Extraction: Edge features are widely used in GNNs to enhance the relationship reasoning between two neighboring nodes. In histopathology images, the category of a nucleus has correlations with its surrounding tissue backgrounds. Thus, we define a graph edge as a local background region centered at the middle point between the two nuclei nodes. To extract an edge feature, we draw a line between two edge-connected nodes, and define an *edge point* as the middle point between the centers of two connected nucleus nodes (see the red crosses in Fig. 3) of the line. The edge feature is sampled from the pixel-wise feature map ($\frac{h}{4} \times \frac{w}{4} \times c$), using the coordinates of the edge point and the bilinear interpolation. Due to the large receptive field obtained through the pixel-wise feature extraction branch, the edge feature contains rich contextual information of a local background region centered at the middle point of two connected nodes.

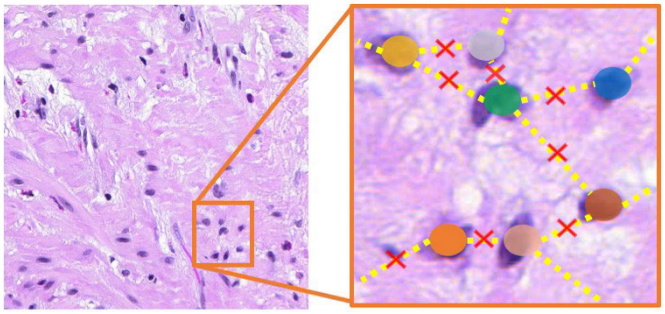


Fig. 3. Visualization of edge points for the feature extraction of edges. An edge point of two connected nuclei nodes is defined as the middle point (red cross) between the position of their centroids.

C. Intra-Nucleus Polygon Structure Learning

The intra-nucleus polygon structure learning module (PSL) is to model a nucleus contour as a polygon, and to learn the shape representation of the nucleus. For a nucleus, we compute its centroid position p_0 and sample n points $p_1, \ldots, p_n$ at the nucleus contour. To better describe the irregular shape, n rays are emitted from the centroid with the same angle interval $\alpha = \frac{2\pi}{n}$, and intersect the boundary at n points that are collected in clockwise order to form a point sequence. To model the positions of contour points relative to the centroid, we insert the centroid at the head of the sequence.

We employ a multi-layer recurrent neural network (RNN) called shape RNN to harvest the shape feature with the point sequence. The input of the RNN is a feature sequence corresponding to the point sequence. The i^{th} element of the feature sequence is defined as $S_i = (i, \gamma_i, L\ PE_i)$ to represent

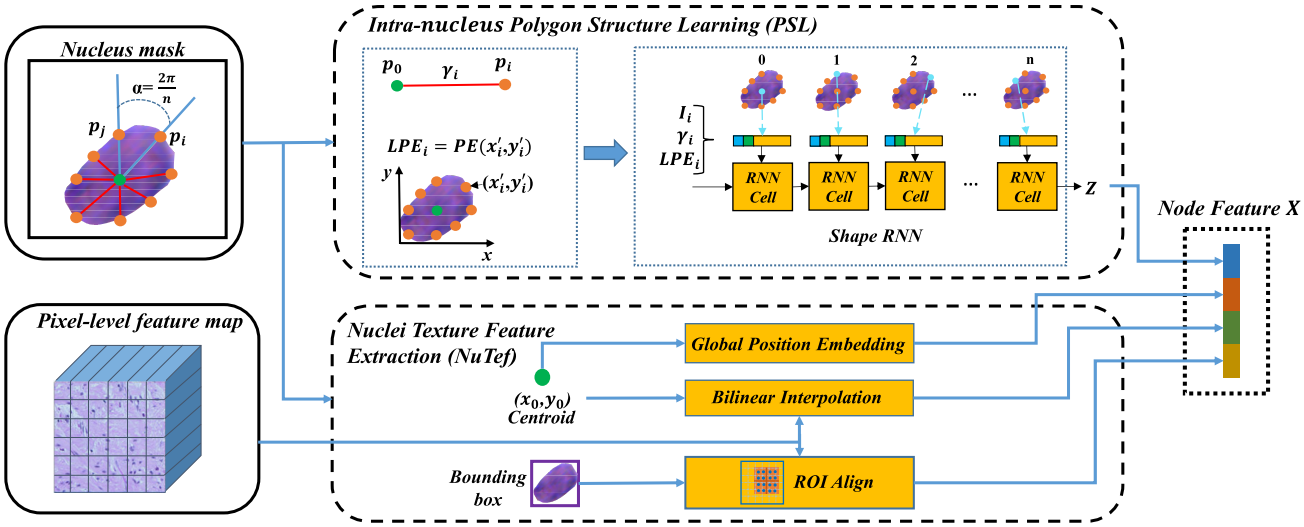


Fig. 4. Node feature extraction process for each nucleus. It consists of two parts: an intra-nucleus Polygon Structure Learning module (PSL) and a Nuclei Texture feature extraction module (NuTef). The PSL module uses a shape RNN to extract the feature Z for a sequence of sampled contour points. $i/\gamma_i/LPE_i$ denotes a point index, the Euclidean distances between p_i and the centroid p_0, the position encoding of p_i in a local coordinate system. The NuTef module computes the textures feature with the bounding box and centroid of a nucleus. The pixel-level feature map is produced by the pixel-wise feature extraction branch as shown in Fig. 2.

the initial feature of each sampled point p_i. i denotes the index of the point sequence, γ_i is the Euclidean distance between a sampled point p_i and the centroid p_0. LPE means the Local Position Encoding. LPE_i denotes the position encoding vector of p_i, and is calculated in a local rectangular coordinate system, which is centered at the bottom-left corner of the bounding box of the nucleus. To run the RNN with the input sequence S, we set the hidden (input) states before the 1^{st} layer of the RNN as $h_i^0 = S_i (0 \le i \le n)$. Then the hidden state of the l^{th} layer and i^{th} input position can be computed as (2):

$$h_i^l = \phi\left(h_{(i-1)}^l W_s^l + h_i^{l-1} W_h^l\right), \tag{2}$$

where W_s^l and W_h^l are the weights of the input and recurrent neurons at the l^{th} hidden layer, h_{i-1}^l is the hidden state of the $(i-1)^{th}$ input position and the $(l)^{th}$ layer. $+$ denotes the elementwise addition. ϕ denotes the ReLU function. If $i-1$ is unavailable (< 0), then $h_{(i-1)}^l W_s^l$ is ignored in (2). The RNN output is based on the hidden state of the final layer:

$$Z = h_n^M W_Z, \tag{3}$$

where M is the layer number of the RNN. W_Z is the weights in the output layer and $Z \in R^{1\times c}$ is the output shape feature for the input nucleus node.

D. Graph Neural Network Architecture

Given a histopathology graph $G = (V, E)$, its initial node features and edge features, we employ a GNN to harvest structure guided representations and to identify types for these nuclei nodes. A GNN usually consists of multiple layers. Each GNN layer aggregates and updates the node features from the previous layer or GNN input. In the aggregating step, the node features in a neighborhood are aggregated into a single feature via a differentiable operator. In the update step, each node feature is updated as the combination of its aggregated neighboring feature and itself.

We implement the GNN using GENeralized Graph Convolution (GENConv) [62] with the DeepGCN [63] structure. GENConv is a graph convolution operator that can deal with edge features. Its key idea is to apply generalized mean-max aggregation functions by keeping the message features positive [62]. The message aggregation and update processes can be formulated as (4) and (5), respectively:

$$a_i = \sigma\left(ReLU\left(X_j + A_{ij} \cdot Y_{ij}\right) + \epsilon\right), \quad j \in \Psi(i), \tag{4}$$

$$X_i = \zeta\left(X_i, a_i\right), \tag{5}$$

where σ is the Softmax aggregations function [62], X_i/X_j denotes the representation of the i^{th}/j^{th} node, $\Psi(i)$ is the set of the neighbor indices of the i^{th} node and ϵ is a small positive constant set to 1e-7. A_{ij} is 1 if the i^{th} and j^{th} nodes are connected by an edge otherwise 0. Y_{ij} is the feature of the edge between the i^{th} and j^{th} nodes. In the update function (5), ζ is a two-layer perceptron using ReLU as the activation functions, and is to enhance X_i with the aggregated feature a_i. To alleviate the vanishing gradient problem, we further utilize the residual connection following DeepGCN as shown in (6):

$$X^{l+1} = \mathcal{F}\left(X^l\right) + X^l, \tag{6}$$

where $\mathcal{F}(\cdot)$ contains a GENConv layer, a batch normalization layer and a ReLU activation function. X^l denotes all the node features X_i^l produced by the l^{th} GNN layer. Finally, with X^L the node representations updated by the last GNN layer, all the nodes are predicted through a classifier described in (7):

$$t = Softmax\left(FC\left(X^L\right)\right). \tag{7}$$

E. Training Scheme

In the proposed framework, the pixel-wise feature extraction branch and the instance-level classification branch are trained simultaneously in an end-to-end manner. The feature extraction branch aims to learn rich texture features by solving a semantic segmentation task with the Dice loss in (8) and the Cross-entropy loss in (9):

$$\mathcal{L}_{\text{CE}} = -\frac{1}{H \times W} \sum_{i=1}^{H \times W} \sum_{q=1}^{Q} y_{i,q}^{s} \log x_{i,q}^{s}, \tag{8}$$

$$\mathcal{L}_{\text{Dice}} = 1 - \frac{2 \times \sum_{i=1}^{H \times W} \sum_{q=1}^{Q} \left(y_{i,q}^{s} \times x_{i,q}^{s} \right) + \varepsilon}{\sum_{i=1}^{H \times W} \sum_{q=1}^{Q} (y_{i,q}^{s} + x_{i,q}^{s}) + \varepsilon}, \tag{9}$$

where x^s is a $HW \times Q$ predicted map of the semantic segmentation task and y^s is the ground truth map. Q is the number of nuclei types, H/W denotes the height/width of a GT or predicted map. ε is a smoothness constant set to 1e-8.

In the instance-level classification branch using the PSL & GSL modules, the classifier predicts category-wise probabilities for each nucleus entity. Cross-entropy loss is widely used in multi-class classification tasks. However, in the nuclei classification task, the distribution of different categories is usually unbalanced and easy samples (such as the nuclei of some tumor cells) could be too dominant to train a robust model. Therefore, we utilize a Focal loss [64] to pay more attentions to hard samples as (10) shows:

$$\mathcal{L}_{\text{Focal}} = -\frac{1}{N} \sum_{i=1}^{N} \sum_{q=1}^{Q} \tau_q (1 - t_{i,q})^{\gamma} y_{i,q}^{o} \log t_{i,q}, \tag{10}$$

where t ($\in R^{N \times Q}$) contains the predicted probabilities for N nuclei objects, y^o is the true labels. γ is a hyper-parameter to make the network concentrate on hard samples. The higher the value of γ, the lower the loss for well-classified examples. τ_q is the weight for each category and is set to the reciprocal of the proportion of the q^{th} class in the training set. The overall objective to train the network is described as (11):

$$\mathcal{L}\left(x^s, t\right) = \mathcal{L}_{\text{Dice}}\left(x^s\right) + \mathcal{L}_{\text{CE}}\left(x^s\right) + \mathcal{L}_{\text{Focal}}\left(t\right), \tag{11}$$

which is composed of two semantic segmentation losses (Dice & Cross-entropy loss) and a classification loss (Focal loss). All three losses are equally weighted.

IV. Experiments

A. Datasets

The proposed framework is evaluated on four nuclei classification datasets: CRC-FFPE, CoNSeP [9], PanNuke [66], MoNuSAC [67]. The CRC-FFPE dataset is an in-house colorectal cancer dataset that consists of 16 patients with 59 H&E stained histopathology tiles of size 1000 × 1000. The images are extracted from the WSIs collected from TCGA [68] and annotated by the pathologists in a local hospital. The nuclei types of the CRC-FFPE dataset include Tumor, Stroma, Immune, Necrosis, and Other. These images are divided into a training set (45 tiles) and a testing set (14 tiles). The CoNSep dataset is a colorectal adenocarcinoma dataset that contains 41 H&E stained images of size 1000 × 1000. The dataset includes 24139 annotated nuclei that are grouped into four categories: Miscellaneous, Inflammatory, Epithelial, and Spindle-shaped. We split the CoNSeP dataset into a training set with 27 images and a testing set with 14 images. The MoNuSAC dataset is a multi-organ dataset, comprising 310 images (209 for training, 101 for testing) of 71 patients. The size of images ranges from 81 × 113 pixels to 1422 × 2162 pixels. The dataset contains four types of organs (breast, kidney, lung, and prostate). The nuclei types of the dataset are: Epithelial, Lymphocytes, Macrophages, and Neutrophils. The PanNuke dataset contains 7899 image tiles of size 256 × 256 of 19 different organs. The images were digitized at 20× or 40× magnification. The nuclei types of the dataset are Inflammatory, Connective, Dead, Epithelial and Neoplastic. We follow the official three-fold data splits of the PanNuke datset. The number of images in Fold 1, Fold 2 and Fold 3 are 2657, 2523 and 2721, respectively.

B. Implementation Details

For the CRC-FFPE and CoNSeP datasets, all the training images are resized to 1024 × 1024. For the MoNuSAC dataset, we crop image patches of size 512 × 512. For the PanNuke dataset, we set the original image size to 256 × 256. We implement the proposed framework with PyTorch [69] and PyTorch Geometric library [70]. The encoder in the pixel-wise extraction branch consists of 4 layers with kernel sizes [3,3,12,3] and channel sizes [64,128,320,512], following the previous work [59]. The encoder is pre-trained on ImageNet [71]. For the GSL module, the GCN model comprises two GENConv [62] layers (L=2) of 64 hidden channels. The neighbor number K is set to 4 for building edges of a graph. For the proposed PSL module, the channel number c of each feature vector for the bounding box, centroid, and positional embedding is 64. The number of hidden layers M of the RNN model is 2 and each layer has 128 hidden units. The number of sampled contour points n in PSL is set to 18. The proposed framework is trained for 100 epochs with an Adam optimizer, an initial learning rate of 1×10^{-4}, and a momentum of 0.9 and 0.99. For all four datasets, γ in the Focal loss is set to 2.

C. Evaluation

Following the previous works [7], [8], [9], we use the F-score F_c [9] to evaluate the nuclei classification methods. The F-score considers the performance of both detection and classification. Given a set of predicted nuclei and a set of ground truth (GT) nuclei, we assign each GT nucleus with its nearest predicted nucleus if their centroids are within 12 pixels, and ensure that no two GT nuclei are assigned to the same predicted nucleus. The predicted nuclei then can be split into the detected, undetected, and wrongly detected ones, whose numbers are denoted as TP_d, FN_d, FP_d, respectively. The classification performance is measured based on TP_d. For one of the categories (for example, the type q), the number of correctly classified nuclei, wrongly classified nuclei, correctly classified nuclei of types other than q, and wrongly classified nuclei instances of types other than type q are denoted as

TABLE I

QUANTITATIVE COMPARISON BETWEEN EXISTING NUCLEI SEGMENTATION & CLASSIFICATION MODELS WITHOUT AND WITH OUR METHOD (+OURS) ON THE CONSEP, MONUSAC, CRC-FFPE AND PANNUKE DATASETS. '*Net*+OURS' MEANS OUR CLASSIFICATION METHOD TAKING THE SEGMENTATION RESULTS OF THE *Net* AS INPUT. $F_c^m, F_c^i, F_c^e, F_c^s, F_c^l, F_c^{ma}, F_c^n, F_c^t, F_c^{st}, F_c^{im}, F_c^{ne}, F_c^o, F_c^c, F_c^d, F_c^{neo}$ REPRESENT THE F-SCORE FOR THE NUCLEI TYPES OF MISCELLANEOUS, INFLAMMATORY, EPITHELIAL, SPINDLE-SHAPED, LYMPHOCYTES, MACROPHAGES, NEUTROPHILS, TUMOR, STROMA, IMMUNE, NECROSIS, OTHER, CONNECTIVE, DEAD, NEOPLASTIC, RESPECTIVELY. F_{avg} IS THE AVERAGE F-SCORE OF THE CATEGORIES IN A DATASET. 'IMP.' IS THE CLASSIFICATION IMPROVEMENT WHEN USING OUR FRAMEWORK

Method	CoNSeP									PanNuke									
	AJI	PQ	F_d	F_{avg}	Imp	F_c^m	F_c^i	F_c^e	F_c^s	AJI	PQ	F_d	F_{avg}	$Imp.$	F_c^i	F_c^c	F_c^d	F_c^e	F_c^{neo}
MCSpatNet [7]	-	-	0.733	0.514	-	0.400	0.537	0.582	0.540	-	-	0.786	0.483	-	0.484	0.473	0.220	0.612	0.629
SRDNet [65]	-	-	-	0.572	-	0.503	0.597	0.641	0.548	-	-	-	**0.529**	-	**0.574**	0.479	0.291	0.648	**0.655**
Triple U-net [39]	0.453	0.412	0.663	0.383	-	0.102	0.570	0.423	0.438	0.583	0.464	0.698	0.381	-	0.392	0.298	**0.297**	0.508	0.412
+Ours	0.453	0.412	0.663	0.421	+3.8%	0.231	0.632	0.476	0.478	0.583	0.464	0.698	0.428	+4.7%	0.436	0.361	0.324	0.564	0.457
Mask2former [40]	0.464	0.482	0.659	0.414	-	0.325	0.461	0.462	0.408	0.616	0.666	0.792	0.480	-	0.400	0.426	0.289	0.668	0.617
+Ours	0.464	0.482	0.659	0.536	+12.2%	0.501	0.548	0.561	0.532	0.616	0.666	0.792	0.519	+3.9%	0.514	0.497	0.256	**0.688**	0.642
TSFD-net [10]	0.458	0.415	0.680	0.439	-	0.120	0.567	0.558	0.513	0.621	0.513	0.748	0.413	-	0.441	0.453	0.102	0.505	0.565
+Ours	0.458	0.415	0.680	0.548	+10.9%	0.462	0.603	0.575	0.554	0.621	0.513	0.748	0.445	+3.2%	0.482	0.477	0.122	0.545	0.601
HoVer-Net [9]	0.544	0.510	0.740	0.549	-	0.430	0.601	0.612	0.552	0.653	0.621	0.787	0.503	-	0.530	0.474	0.260	0.618	0.632
+Ours	0.544	0.510	0.740	**0.595**	+4.6%	**0.510**	**0.632**	**0.646**	**0.592**	0.653	0.621	0.787	0.528	+2.5%	0.541	**0.507**	0.263	0.684	0.643
Method	MoNuSAC									CRC-FFPE									
	AJI	PQ	F_d	F_{avg}	Imp	F_c^e	F_c^l	F_c^{ma}	F_c^n	AJI	PQ	F_d	F_{avg}	$Imp.$	F_c^t	F_c^{st}	F_c^{im}	F_c^{ne}	F_c^o
MCSpatNet [7]	-	-	0.828	0.651	-	0.698	0.753	**0.517**	0.636	-	-	0.752	0.278	-	0.547	0.196	0.307	0.126	0.212
Triple U-net [39]	0.408	0.543	0.638	0.464	-	0.557	0.637	0.276	0.386	0.510	0.547	0.684	0.196	-	0.458	0.101	0.212	0.079	0.129
+Ours	0.408	0.543	0.638	0.536	+7.2%	0.599	0.768	0.302	0.477	0.510	0.547	0.684	0.255	+5.9%	0.479	0.134	0.320	0.111	0.231
Mask2former [40]	0.577	0.568	0.746	0.559	-	0.682	0.762	0.349	0.446	0.451	0.491	0.581	0.212	-	0.299	0.164	0.278	0.101	0.218
+Ours	0.577	0.568	0.746	0.603	+4.4%	0.747	0.786	0.377	0.502	0.451	0.491	0.581	0.252	+4%	0.399	0.168	0.312	0.121	0.263
TSFD-net [10]	0.461	0.499	0.735	0.350	-	0.544	0.738	0.091	0.027	0.478	0.503	0.549	0.182	-	0.308	0.110	0.201	0.098	0.194
+Ours	0.461	0.499	0.735	0.435	+8.5%	0.621	0.788	0.210	0.124	0.478	0.503	0.549	0.217	+2.5%	0.341	0.145	0.278	0.101	0.223
HoVer-Net [9]	0.599	0.613	0.748	0.604	-	0.754	0.806	0.394	0.463	0.603	0.636	0.743	0.245	-	0.512	0.180	0.290	0.096	0.148
+Ours	0.599	0.613	0.748	**0.692**	+8.8%	**0.799**	**0.846**	0.383	**0.742**	0.603	0.636	0.743	**0.353**	+10.8%	**0.548**	**0.196**	**0.418**	**0.159**	**0.446**

TP_c, FP_c, TN_c and FN_c, respectively. The F-score of type q is computed as (12):

$$F_c^q = \frac{2\left(TP_c + TN_c\right)}{\left[\begin{array}{c} 2\left(TP_c + TN_c\right) + \alpha_0 FP_c + \alpha_1 FN_c \\ +\alpha_2 FP_d + \alpha_3 FN_d \end{array}\right]}, \tag{12}$$

where $\alpha_0 = 2$, $\alpha_1 = 2$, $\alpha_2 = 1$ and $\alpha_3 = 1$. The average F-score of all categories in a dataset is reported as F_{avg} and can be viewed as a general metric of classification performance.

To show the detection and segmentation results of previous works, we evaluate the nuclei segmentation metrics: Aggregated Jaccard Index (AJI) [72], Panoptic Quality (PQ) [73], Detection Quality F_d [9]. AJI is an extension of the global Jaccard index and measures the overlapping areas of multiple objects and is recognized as an object-level criterion for segmentation evaluation. PQ is another metric for accurate quantification of detection and segmentation. It is defined as $\mathcal{PQ} = \frac{|TP|}{|TP|+\frac{1}{2}|FP|+\frac{1}{2}|FN|} \times \frac{\sum_{(x,y)\in TP} \mathrm{IoU}(x,y)}{|TP|}$. The first part of PQ is the detection quality F_d. Each prediction-GT pairs are matched to be unique if their IoU(x,y) is larger than 0.5. The predictions and GT are split into matched pairs (TP), unmatched GT (FN) and unmatched predictions (FP). The detection quality F_d then is defined as the F_1 score for instance detection. The second part of PQ is the segmentation quality which can be interpreted as how close each correctly detected instance is to its matched GT.

D. Comparison With the Existing Methods

We compare our proposed approach with the existing classification methods SRDNet [65], HoVer-Net [9], Triple U-net [39], MCSpatNet [7], TSFD-net [10], and Mask2former [40]. Among them, SRDNet, HoVer-Net, MCSpatNet, and Mask2former support nuclei classification. TSFD-net is a semantic segmentation method and Triple U-net is an instance segmentation method. For SRDNet, HoVer-Net, MCSpatNet and Mask2former, we directly compare the instance classification performance using their original settings. The nuclei classification F-scores for SRDNet are from its original paper. For TSFD-net, we extract the instance classification results using its semantic segmentation outputs and instance segmentation outputs. For Triple U-net, an extra 1×1 convolution is added as the classification layer. Note that we aim at solving the classification task in this paper. To fairly compare the classification performance among existing methods and ours, we propose an evaluation setting 'instance-ground truth' where the GT maps of nuclei instance segmentation are accessible to all these methods during the testing stage. In this setting, these methods excluding Mask2former use the GT segmentation maps to replace their own predicted segmentation results for classifying nuclei. For Mask2former, we find the nearest predicted instance mask for each instance GT, and assign the predicted cell type to the instance GT. Since each method adopts the segmentation results of the same quality, the 'instance-ground truth' setting provides a fair comparison of nuclei classification. We also report the results on a typical setting 'instance-prediction' where the segmentation GTs are not accessible and each method needs to predict its own segmentation results.

As TABLE I shows, in the 'instance-prediction' setting the proposed framework outperforms the existing methods on all 4 types in the CoNSeP dataset, all 5 types in the CRC-FFPE dataset, 3 types in the MoNuSAC dataset and 2 types in the PanNuke dataset. '*Model* + Ours' denotes our proposed

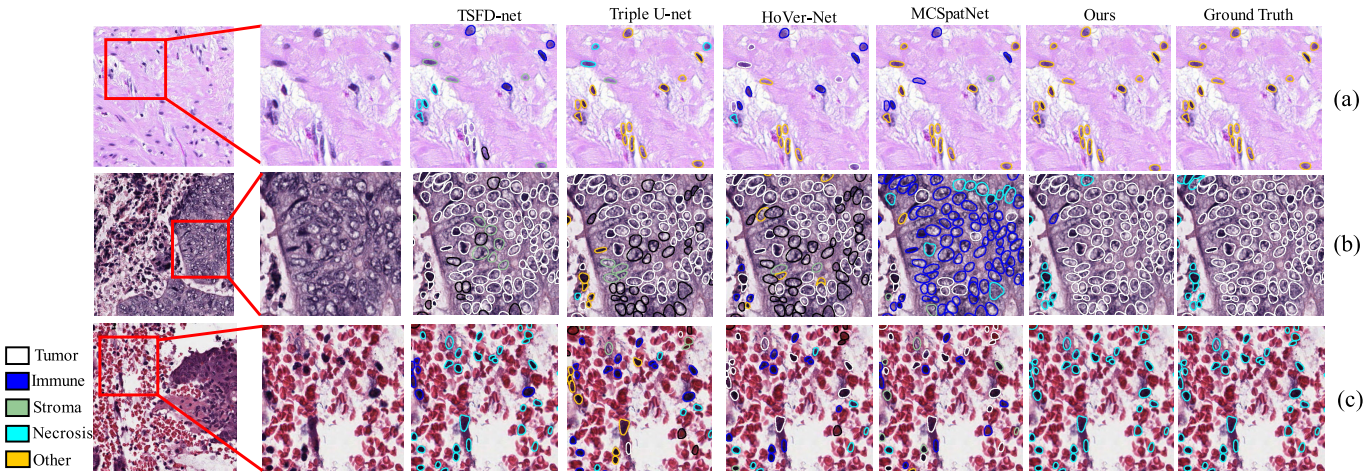


Fig. 5. Visualized classification results on the CRC-FFPE dataset and the 'instance-ground truth' setting where the GTs of nuclei instance segmentation are accessible. We compare the classification performance between our framework and the existing methods.

TABLE II

QUANTITATIVE COMPARISON BETWEEN EXISTING METHODS AND OURS FOR NUCLEI CLASSIFICATION ON THE CONSEP, MONUSAC, CRC-FFPE AND PANNUKE DATASETS IN THE 'INSTANCE-GROUND TRUTH' SETTING. THE GROUND TRUTH OF NUCLEI INSTANCE SEGMENTATION IS ACCESSIBLE FOR ALL THE METHODS IN THE INFERENCE STAGE

Method	CoNSeP					CRC-FFPE					
	F_{avg}	F_c^m	F_c^i	F_c^e	F_c^s	F_{avg}	F_c^t	F_c^{st}	F_c^{im}	F_c^{ne}	F_c^o
HoVer-Net [9]	0.711	0.585	0.656	0.872	0.730	0.350	0.737	0.240	0.290	0.153	0.327
Mask2former [40]	0.523	0.435	0.522	0.573	0.565	0.243	0.355	0.201	0.257	0.166	0.237
Triple U-net [39]	0.438	0.170	0.399	0.684	0.501	0.289	0.599	0.202	0.344	0.099	0.202
TSFD-net [10]	0.632	0.502	0.635	0.822	0.570	0.221	0.378	0.176	0.222	0.100	0.233
MCSpatNet [7]	0.695	0.562	0.624	0.863	0.730	0.341	0.772	0.190	0.372	0.071	0.301
Ours	**0.777**	**0.726**	**0.685**	**0.890**	**0.807**	**0.448**	**0.804**	**0.271**	**0.449**	**0.184**	**0.534**
Method	MoNuSAC					PanNuke					
	F_{avg}	F_c^e	F_c^l	F_c^{ma}	F_c^n	F_{avg}	F_c^i	F_c^c	F_c^d	F_c^{ep}	F_c^{ne}
HoVer-Net [9]	0.701	0.794	0.910	0.612	0.486	0.529	0.525	0.610	0.278	0.636	0.598
Mask2former [40]	0.680	0.810	0.868	0.467	0.575	0.527	0.463	0.478	0.307	0.698	0.687
Triple U-net [39]	0.530	0.629	0.747	0.331	0.412	0.396	0.412	0.278	0.304	0.555	0.432
TSFD-net [10]	0.605	0.695	0.902	0.435	0.388	0.418	0.423	0.210	**0.378**	0.602	0.478
MCSpatNet [7]	0.828	0.955	0.963	0.683	0.712	0.574	0.549	0.576	0.269	0.702	0.772
Ours	**0.919**	**0.967**	**0.972**	**0.862**	**0.873**	**0.621**	**0.607**	**0.622**	0.253	**0.823**	**0.799**

framework taking the segmentation predictions of the *Model* as input. 'Imp.' is the classification F1-score improvement over each previous method when using our classification framework. The results display that our method can significantly improve the previous nuclei classification methods by 3.8%-12.2%, 4.4%-8.8%, 2.5%-10.8%, 2.5%-3.9% average F-score on the CoNSeP, MoNuSAC, CRC-FFPE, and PanNuke datasets, respectively. In TABLE II, on the 'instance-ground truth' setting, our proposed framework achieves the highest F-score on all 4 types in the CoNSeP dataset, all 4 types in the MoNuSAC dataset, all 5 types in the CRC-FFPE dataset and 4 types in the PanNuke dataset. The proposed method outperforms the second-best model by 6.6%, 9.1%, 9.8%, and 4.7% average F-score on the four datasets. The results indicate that our proposed framework has the ability to significantly improve the nuclei classification performance for existing methods.

Fig. 5 visualizes the nuclei classification results of some existing methods and ours on the CRC-FFPE dataset. The visual results are obtained on the 'instance-ground truth' setting to compare only the classification performance among these approaches. In Fig. 5(a), our method can accurately identify sparsely-distributed nuclei, since a GCN is utilized to propagate contextual information among even remote nuclei. In Fig. 5(b), the proposed framework shows its advantage of classifying densely-distributed nuclei of the same type. Most of these nuclei are surrounded by similar background regions, which can be well described by the edge representations and guide the nuclei classification in our proposed method.

TABLE III

ABLATION STUDY ON THE CONSEP DATASET AND USING THE GROUND TRUTH AS SEGMENTATION MAP IN THE INFERENCE STAGE

Method	CoNSeP				
	F_{avg}	F_c^m	F_c^i	F_c^e	F_c^s
(a) Baseline	0.437	0.187	0.416	0.721	0.422
(b) +GCN	0.651	0.437	0.575	0.869	0.714
(c) +EdgeFeat	0.690	0.580	0.592	0.879	0.704
(d) Ours	**0.777**	**0.726**	**0.685**	**0.890**	**0.807**

E. Ablation Study

We perform the ablation study on the CoNSeP dataset and the 'instance-ground truth' setting. In TABLE III, (a)'Baseline'

TABLE IV
COMPUTATIONAL EFFICIENCY ON WHOLE SLIDE IMAGES. INFERENCE TIME IS MEASURED AS THE AVERAGE TIME OF INFERRING TEN WHOLE SLIDE IMAGES

Method	#Para. (M)	Infer Time (s)	Storage Size (Mb)
Hover-net	33.60	2584	144
Ours	37.43	513	319

consists of the pixel-wise feature extraction branch and a simple nuclei feature extraction module without PSL and GSL. (b)'+GCN' is the baseline using a GCN without edge features. (c)'+EdgeFeat' denotes the baseline using a GCN with edge features. (d)'Ours' is the proposed framework with the PSL & GSL modules. Comparing (a) to (b), using the GCN significantly improves the classification performance by around 21.4% average F-score. That shows the powerful ability of the graph network in modeling relationships. Comparing (c) to (b) shows that the proposed edge features improve the average F-score by 3.9%. It indicates that the background information provided by the proposed edge feature helps better identify the nuclei types. Comparing (d) to (c) suggests that our proposed PSL module leads to a great increase of 8.7% average F-score. Overall, the proposed framework surpasses the baseline by 34.0% average F-score when the segmentation GT is available.

1) Computational Efficiency: We evaluate our proposed classification framework on a machine with Ubuntu 20.04, an NVIDIA-A6000 GPU with 48 GB memory, and Intel(R) Xeon(R) W-2235 CPU with 64 GB memory. Training our proposed method cost 8 hours to 2 days for these four datasets. The GPU memory cost and inference time for an image patch of size 1000×1000 are about 6 GB and 1.97s. To evaluate the feasibility of real-world applications of our proposed framework, we assessed the average parameter count (#Para), inference time (Infer Time), and storage size on ten whole slide images (WSIs) in TABLE IV. These WSIs have an average size of 53672×74692. The WSIs are randomly selected from The Cancer Genome Atlas (TCGA) database. They have an average size of 53672×74692. The numerical results of 'Ours' in TABLE IV do not include the trained segmentation model used by our method. In comparison to Hover-net, our framework increases about 20% inference time and 300 MB hard disk storage. We argue that the extra computational overhead is acceptable, considering the low cost of hard disk and the significant improvement in performance.

F. Investigation of Hyper-Parameters

In TABLE V, we study the selection of two hyper-parameters on the CoNSep dataset. One is the neighbor number K, which determines how many neighbors are connected to a nucleus in the graph. The other one is the number of sampled contour points n, which determines how fine-grained the nucleus shape is in the PSL module. All the experiments are on the 'instance-ground truth' setting. As TABLE V shows, setting K to 4 achieves the highest F_{avg} while setting K to 3/6 results in a drop of 2.5%/5.0% F-score. Larger K means more connected neighbors and larger weights for the contextual features. It may be due to that too much context information makes the model pay less attention to the original texture or shape features of predicted nuclei. Thus, setting K to a moderate value mostly benefits our method. Setting n to 18 obtains the best F_{avg} while setting n to 36 causes a drop of 3.8% F_{avg}. It may because the newly sampled points are not distinct enough and could act as noises to cause overfitting.

TABLE V
HYPER-PARAMETERS INVESTIGATION OF THE NEIGHBOR NUMBER K AND THE NUMBER OF SAMPLED CONTOUR POINTS n ON THE CONSEP DATASET

n=18	F_{avg}	F_c^m	F_c^i	F_c^e	F_c^s
K=3	0.752	0.785	0.577	0.915	0.734
K=4	0.777	0.726	0.685	0.890	0.807
K=6	0.727	0.721	0.591	0.895	0.704
K=4	F_{avg}	F_c^m	F_c^i	F_c^e	F_c^s
n=9	0.727	0.741	0.577	0.912	0.681
n=18	0.777	0.726	0.685	0.890	0.807
n=36	0.739	0.762	0.576	0.914	0.705

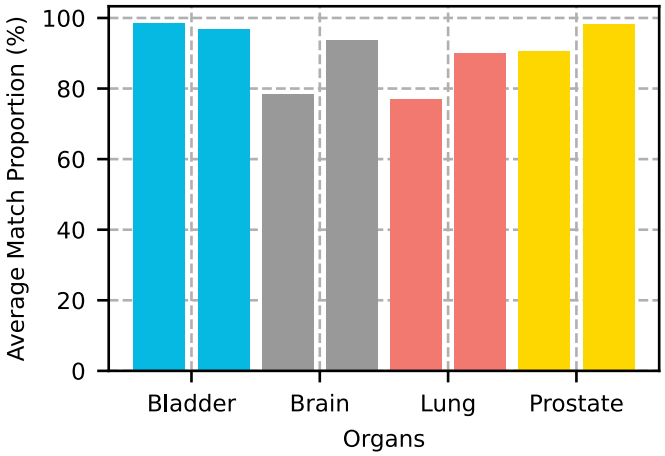


Fig. 6. Average match proportion of eight WSIs for four organs. Each bar represents the result of a whole-slide image.

G. Visual Evaluation on Whole-Slide Images

To show the effectiveness of our method in the real applications of in-the-wild whole-slide images (WSIs), we collected eight WSIs of four organs (Bladder, Brain, Lung, Prostate) from the Cancer Genome Atlas (TCGA) [68] database. Each organ contains two WSIs. For each WSI, we first obtain the nuclei segmentation results using a Hover-net [9] model trained on the PanNuke dataset. Then, we predict cell types for the WSI using a sliding window manner with our PanNuke-trained model. The predicted cell classes include Inflammatory, Connective, Dead, Epithelial, and Neoplastic. The window size and stride are set to 256×256 and 128, respectively.

To reduce the workload of pathologists in visually assessing the entire slide, we randomly cropped six regions of size 2000×2000 from each slide. For each region, pathologists are only required to roughly estimate the proportion (ranging from 0% to 100%) of predicted cells whose labels match with the categories estimated by the pathologists. Subsequently, we define the match proportion of a WSI as the average of the match proportions obtained from the six cropped regions.

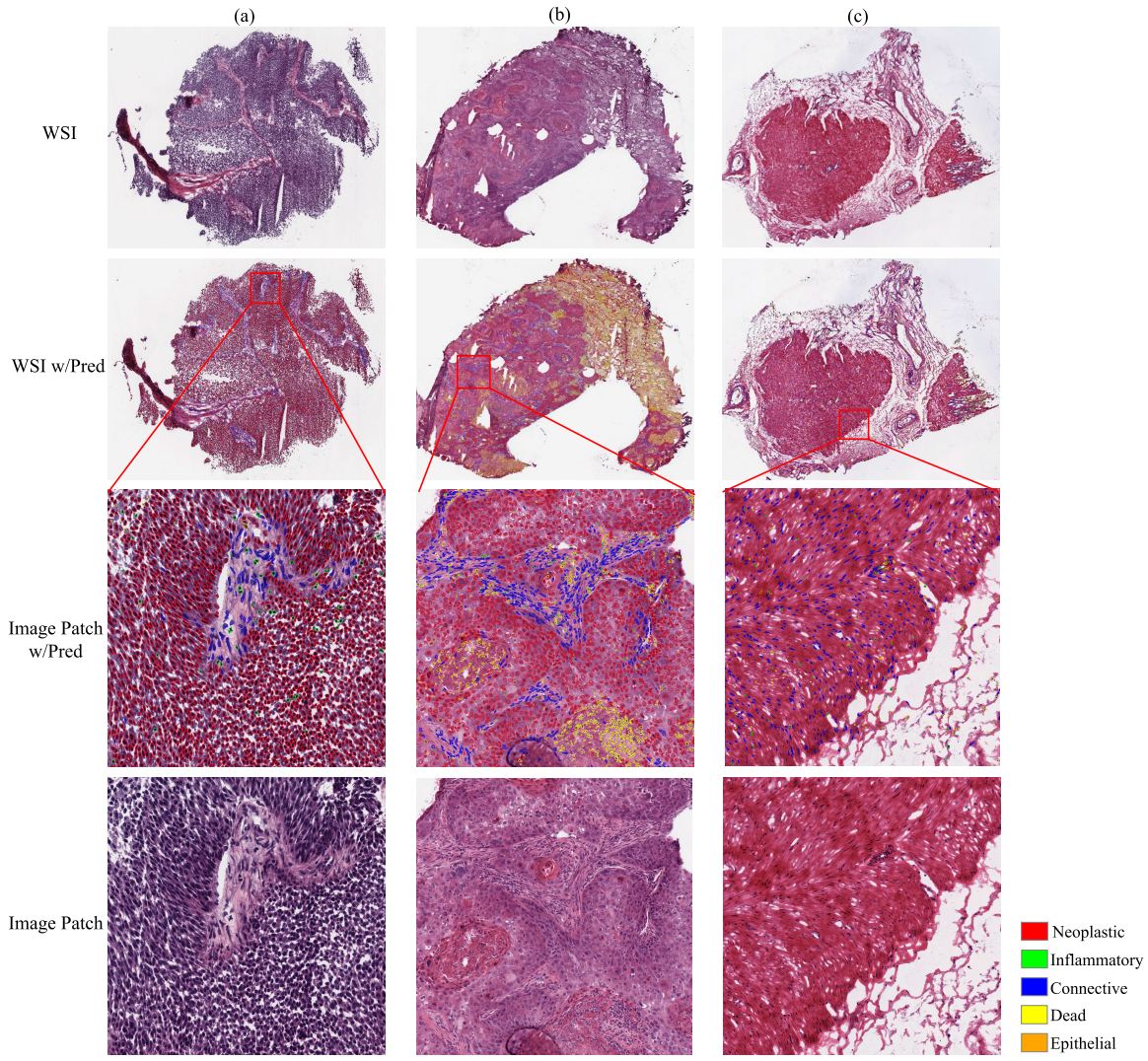


Fig. 7. Visual results on whole-slide images (WSIs). The results are predicted by our framework trained on the PanNuke dataset. The WSIs are randomly selected from The Cancer Genome Atlas (TCGA) [68] database. 'w/Pred' means that the WSI or cropped image patches are covered with the nuclei classification results predicted by our framework.

Fig. 6 shows the average match proportion of the collected WSIs. Each bar in Fig. 6 represents the result of a WSI. As shown in Fig. 6, our framework achieves a match proportion of over 75% for each evaluated WSIs, and the match proportions of 90% on the organs Bladder and Prostate. The results of visual evaluation conducted by the pathologists show the practical value of our framework in real applications. In Fig. 7, we present some visual examples of the predictions from our framework on these WSIs.

V. Discussion

Regarding the number of connections in the graph, a performance degradation is observed in TABLE V. It should be taken into account that in a histopathology image, not every pair of cells has correlations in its types. According to relevant medical theories [51], [74], [75], the cells close to each other are more likely to belong to the same category or have some correlations in their types. Therefore, we determine to connect the nucleus nodes that are close in distance. However, if we increase the number of connections K, the distant and less relevant nuclei could be connected by edges. These unreliable connections may affect the GNN training and degrade the performance.

As described in the general workflow (Fig. 2), our methodology requires a trained nuclei segmentation model to provide binary-class instance segmentation maps. Any trained model that is able to output the boundary or mask of each nucleus instance in an input image, can be used by our method. Note that the proposed approach does not require the segmentation model to label fine-grained cell types. As TABLE I shows, our method can work with four existing segmentation models (Triplet U-net, Mask2former, TSFD-net, Hovernet), achieving enhanced performance. Actually, in some cases, Triplet U-net and TSFD-net are not so effective or successful in segmenting nuclei, but our method still improves the nuclei classification when integrating with the two models. For nuclei detection models, since they do not produce the boundaries of nuclei which are required by the PSL module, it is not feasible to take them as the starting point of our methodology.

VI. Conclusion

In this paper, we aim to solve a challenging task of automatic nuclei classification for H&E stained multi-organ histopathology images. First, we propose a novel structure-embedded nuclei classification framework. Second, we build an inter-nuclei graph structure learning module to capture rich contextual information and short-long range correlations among nuclei. Third, we develop an intra-nuclei polygon structure learning module for harvesting better shape representations of a nucleus using a recurrent neural network. The experimental results suggest that both our overall framework and the proposed modules can significantly surpass the existing methods. In the future, it would be meaningful to extend our framework to a unified graph-based nuclei detection and classification model and apply the model to more cancer types of various organs.

References

[1] H. Xu et al., "Vision transformers for computational histopathology," *IEEE Rev. Biomed. Eng.*, vol. 17, pp. 63–79, 2024.

[2] U. Zidan, M. M. Gaber, and M. M. Abdelsamea, "SwinCup: Cascaded Swin transformer for histopathological structures segmentation in colorectal cancer," *Expert Syst. Appl.*, vol. 216, Apr. 2023, Art. no. 119452.

[3] C. Lu et al., "Nuclear shape and orientation features from H&E images predict survival in early-stage estrogen receptor-positive breast cancers," *Lab. Invest.*, vol. 98, no. 11, pp. 1438–1448, Nov. 2018.

[4] S. Tabibu, P. Vinod, and C. Jawahar, "Pan-renal cell carcinoma classification and survival prediction from histopathology images using deep learning," *Sci. Rep.*, vol. 9, no. 1, pp. 1–9, 2019.

[5] M. G. Ertosun and D. L. Rubin, "Automated grading of gliomas using deep learning in digital pathology images: A modular approach with ensemble of convolutional neural networks," in *Proc. AMIA Annu. Symp.*, 2015, p. 1899.

[6] H. D. Couture et al., "Image analysis with deep learning to predict breast cancer grade, ER status, histologic subtype, and intrinsic subtype," *NPJ Breast Cancer*, vol. 4, no. 1, pp. 1–8, Sep. 2018.

[7] S. Abousamra et al., "Multi-class cell detection using spatial context representation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 4005–4014.

[8] T. N. N. Doan, B. Song, T. T. L. Vuong, K. Kim, and J. T. Kwak, "SONNET: A self-guided ordinal regression neural network for segmentation and classification of nuclei in large-scale multi-tissue histology images," *IEEE J. Biomed. Health Informat.*, vol. 26, no. 7, pp. 3218–3228, Jul. 2022.

[9] S. Graham et al., "Hover-net: Simultaneous segmentation and classification of nuclei in multi-tissue histology images," *Med. Image Anal.*, vol. 58, Dec. 2019, Art. no. 101563.

[10] T. Ilyas, Z. I. Mannan, A. Khan, S. Azam, H. Kim, and F. De Boer, "TSFD-Net: Tissue specific feature distillation network for nuclei segmentation and classification," *Neural Netw.*, vol. 151, pp. 1–15, Jul. 2022.

[11] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 1–14. [Online]. Available: https://openreview.net/forum?id=SJU4ayYgl

[12] M. Sureka, A. Patil, D. Anand, and A. Sethi, "Visualization for histopathology images using graph convolutional neural networks," in *Proc. IEEE 20th Int. Conf. Bioinf. Bioengineering (BIBE)*, Oct. 2020, pp. 331–335.

[13] Y. Zhao et al., "Predicting lymph node metastasis using histopathological images based on multiple instance learning with deep graph convolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2020, pp. 4837–4846.

[14] V. Anklin et al., "Learning whole-slide segmentation from inexact and incomplete labels using tissue graphs," in *Proc. Int. Conf. Med. Image Comput. Comput. Assist. Intervent.* Cham, Switzerland: Springer, 2021, pp. 636–646.

[15] G. Jaume et al., "Quantifying explainers of graph neural networks in computational pathology," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jul. 2021, pp. 8106–8116.

[16] S. Liu, P. A. Mundra, and J. C. Rajapakse, "Features for cells and nuclei classification," in *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, Aug. 2011, pp. 6601–6604.

[17] H. Sharma et al., "A multi-resolution approach for combining visual information using nuclei segmentation and classification in histopathological images," in *Proc. 10th Int. Conf. Comput. Vis. Theory Appl.*, 2015, pp. 37–46.

[18] L. Zhang, L. Lu, I. Nogues, R. M. Summers, S. Liu, and J. Yao, "DeepPap: Deep convolutional networks for cervical cell classification," *IEEE J. Biomed. Health Informat.*, vol. 21, no. 6, pp. 1633–1643, Nov. 2017.

[19] S. H. S. Basha et al., "RCCNet: An efficient convolutional neural network for histological routine colon cancer nuclei classification," in *Proc. 15th Int. Conf. Control, Autom., Robot. Vis. (ICARCV)*, Nov. 2018, pp. 1222–1227.

[20] K. Zormpas-Petridis, H. Failmezger, S. E. A. Raza, I. Roxanis, Y. Jamin, and Y. Yuan, "Superpixel-based conditional random fields (SuperCRF): Incorporating global and local context for enhanced deep learning in melanoma histopathology," *Frontiers Oncol.*, vol. 9, p. 1045, Oct. 2019.

[21] S. Graham et al., "One model is all you need: Multi-task learning enables simultaneous histology image segmentation and classification," *Med. Image Anal.*, vol. 83, Jan. 2023, Art. no. 102685.

[22] K. Sirinukunwattana et al., "Locality sensitive deep learning for detection and classification of nuclei in routine colon cancer histology images," *IEEE Trans. Med. Imag.*, vol. 35, no. 5, pp. 1196–1206, May 2016.

[23] Y. Feng, L. Zhang, and Z. Yi, "Breast cancer cell nuclei classification in histopathology images using deep neural networks," *Int. J. Comput. Assist. Radiol. Surgery*, vol. 13, no. 2, pp. 179–191, Feb. 2018.

[24] F. Hörst et al., "CellViT: Vision transformers for precise cell segmentation and classification," 2023, *arXiv:2306.15350*.

[25] Y. Chen, J. Feng, J. Liu, B. Pang, D. Cao, and C. Li, "Detection and classification of lung cancer cells using Swin transformer," *J. Cancer Therapy*, vol. 13, no. 7, pp. 464–475, 2022.

[26] S. Chen, C. Ding, and D. Tao, "Boundary-assisted region proposal networks for nucleus segmentation," in *Proc. Med. Image Comput. Comput. Assist. Intervent.-MICCAI 2020, 23rd Int. Conf.*, Lima, Peru. Cham, Switzerland: Springer, 2020, pp. 279–288.

[27] H.-Y. Zhou et al., "SSMD: Semi-supervised medical image detection with adaptive consistency and heterogeneous perturbation," *Med. Image Anal.*, vol. 72, Aug. 2021, Art. no. 102117.

[28] J. Huang, H. Li, W. Sun, X. Wan, and G. Li, "Prompt-based grouping transformer for nucleus detection and classification," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assisted Intervent.* Cham, Switzerland: Springer, 2023, pp. 569–579.

[29] S. Chen, C. Ding, M. Liu, J. Cheng, and D. Tao, "CPP-Net: Context-aware polygon proposal network for nucleus segmentation," *IEEE Trans. Image Process.*, vol. 32, pp. 980–994, 2023.

[30] H. Höfener, A. Homeyer, N. Weiss, J. Molin, C. F. Lundström, and H. K. Hahn, "Deep learning nuclei detection: A simple approach can deliver state-of-the-art results," *Computerized Med. Imag. Graph.*, vol. 70, pp. 43–52, Dec. 2018.

[31] J. Huang, H. Li, X. Wan, and G. Li, "Affine-consistent transformer for multi-class cell nuclei detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 21384–21393.

[32] X. Yu et al., "Diffusion-based data augmentation for nuclei image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Cham, Switzerland: Springer, 2023, pp. 592–602.

[33] J. Ma et al., "The multi-modality cell segmentation challenge: Towards universal solutions," 2023, *arXiv:2308.05864*.

[34] W. Lou et al., "Multi-stream cell segmentation with low-level cues for multi-modality images," in *Proc. Competitions Neural Inf. Process. Syst.*, 2023, pp. 1–10.

[35] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2961–2969.

[36] K. Roy, S. Saha, D. Banik, and D. Bhattacharjee, "Nuclei-net: A multi-stage fusion model for nuclei segmentation in microscopy images," *Innov. Syst. Softw. Eng.*, Dec. 2023, doi: 10.1007/s11334-023-00537-y.

[37] D. Liu, D. Zhang, Y. Song, H. Huang, and W. Cai, "Panoptic feature fusion net: A novel instance segmentation paradigm for biomedical and biological images," *IEEE Trans. Image Process.*, vol. 30, pp. 2045–2059, 2021.

[38] Y. Zhou, O. F. Onder, Q. Dou, E. Tsougenis, H. Chen, and P.-A. Heng, "CIA-Net: Robust nuclei instance segmentation with contour-aware information aggregation," in *Proc. Int. Conf. Inf. Process. Med. Imag.*, Hong Kong. Cham, Switzerland: Springer, 2019, pp. 682–693.

[39] B. Zhao et al., "Triple U-net: hematoxylin-aware nuclei segmentation with progressive dense feature aggregation," *Med. Image Anal.*, vol. 65, Oct. 2020, Art. no. 101786.

[40] B. Cheng, I. Misra, A. G. Schwing, A. Kirillov, and R. Girdhar, "Masked-attention mask transformer for universal image segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 1290–1299.

[41] Y. Zhou, S. Graham, N. Alemi Koohbanani, M. Shaban, P.-A. Heng, and N. Rajpoot, "CGC-net: Cell graph convolutional network for grading of colorectal cancer histology images," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW)*, Oct. 2019, pp. 388–398.

[42] J. Wang, R. J. Chen, M. Y. Lu, A. Baras, and F. Mahmood, "Weakly supervised prostate TMA classification via graph convolutional networks," in *Proc. IEEE 17th Int. Symp. Biomed. Imag. (ISBI)*, Apr. 2020, pp. 239–243.

[43] K. Ding, Q. Liu, E. Lee, M. Zhou, A. Lu, and S. Zhang, "Feature-enhanced graph networks for genetic mutational prediction using histopathological images in colon cancer," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Cham, Switzerland: Springer, 2020, pp. 294–304.

[44] L. Studer, J. Wallau, H. Dawson, I. Zlobec, and A. Fischer, "Classification of intestinal gland cell-graphs using graph neural networks," in *Proc. 25th Int. Conf. Pattern Recognit. (ICPR)*, Jan. 2021, pp. 3636–3643.

[45] R. J. Chen et al., "Pathomic fusion: An integrated framework for fusing histopathology and genomic features for cancer diagnosis and prognosis," *IEEE Trans. Med. Imag.*, vol. 41, no. 4, pp. 757–770, Apr. 2022.

[46] T. Hassan, S. Javed, A. Mahmood, T. Qaiser, N. Werghi, and N. Rajpoot, "Nucleus classification in histology images using message passing network," *Med. Image Anal.*, vol. 79, Jul. 2022, Art. no. 102480.

[47] Y. Lee et al., "Derivation of prognostic contextual histopathological features from whole-slide images of tumours via graph deep learning," *Nature Biomed. Eng.*, vol. 6, pp. 1–15, Aug. 2022.

[48] P. Pati et al., "Hierarchical graph representations in digital pathology," *Med. Image Anal.*, vol. 75, Jan. 2022, Art. no. 102264.

[49] Y. Zheng et al., "A graph-transformer for whole slide image classification," *IEEE Trans. Med. Imag.*, vol. 41, no. 11, pp. 3003–3015, Nov. 2022.

[50] J. Shi et al., "A structure-aware hierarchical graph-based multiple instance learning framework for pT staging in histopathological image," *IEEE Trans. Med. Imag.*, vol. 42, no. 10, pp. 3000–3011, Oct. 2023.

[51] S. Javed et al., "Cellular community detection for tissue phenotyping in colorectal cancer histology images," *Med. Image Anal.*, vol. 63, Jul. 2020, Art. no. 101696.

[52] B. Aygüneş et al., "Graph convolutional networks for region of interest classification in breast histopathology," in *Proc. SPIE*, vol. 11320, 2020, pp. 134–141.

[53] C. Gunduz, B. Yener, and S. H. Gultekin, "The cell graphs of cancer," *Bioinformatics*, vol. 20, no. suppl_1, pp. i145–i151, Aug. 2004.

[54] L. Castrejon, K. Kundu, R. Urtasun, and S. Fidler, "Annotating object instances with a polygon-RNN," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5230–5238.

[55] U. Schmidt, M. Weigert, C. Broaddus, and G. Myers, "Cell detection with star-convex polygons," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent. (MICCAI)*. Cham, Switzerland: Springer, 2018, pp. 265–273.

[56] W. Shen, X. Wang, Y. Wang, X. Bai, and Z. Zhang, "DeepContour: A deep convolutional feature learned by positive-sharing loss for contour detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3982–3991.

[57] H. Chen, X. Qi, L. Yu, and P. Heng, "DCAN: Deep contour-aware networks for accurate gland segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2487–2496.

[58] C. Huang, H. Li, Y. Xie, Q. Wu, and B. Luo, "PBC: Polygon-based classifier for fine-grained categorization," *IEEE Trans. Multimedia*, vol. 19, no. 4, pp. 673–684, Apr. 2017.

[59] M.-H. Guo, C.-Z. Lu, Z.-N. Liu, M.-M. Cheng, and S.-M. Hu, "Visual attention network," 2022, *arXiv:2202.09741*.

[60] T. Y. Lin, P. Dollàr, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 2117–2125.

[61] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–12.

[62] G. Li, C. Xiong, A. Thabet, and B. Ghanem, "DeeperGCN: All you need to train deeper GCNs," 2020, *arXiv:2006.07739*.

[63] G. Li et al., "DeepGCNs: Making GCNs go as deep as CNNs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 6, pp. 6923–6939, Jun. 2023.

[64] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.

[65] S. Xiao, A. Qu, H. Zhong, and P. He, "A scale and region-enhanced decoding network for nuclei classification in histology image," *Biomed. Signal Process. Control*, vol. 83, May 2023, Art. no. 104626.

[66] J. Gamper et al., "PanNuke dataset extension, insights and baselines," 2020, *arXiv:2003.10778*.

[67] R. Verma et al., "MoNuSAC2020: A multi-organ nuclei segmentation and classification challenge," *IEEE Trans. Med. Imag.*, vol. 40, no. 12, pp. 3413–3423, Dec. 2021.

[68] J. N. Weinstein et al., "The cancer genome atlas pan-cancer analysis project," *Nat. Genet.*, vol. 45, no. 10, pp. 1113–1120, 2013.

[69] A. Paszke et al., "Automatic differentiation in Pytorch," in *Proc. Adv. Neural Inf. Process. Syst. Workshop*, 2017, pp. 1–4.

[70] M. Fey and J. E. Lenssen, "Fast graph representation learning with Pytorch geometric," in *Proc. ICLR Workshop Represent. Learn. Graphs Manifolds*, 2019, pp. 1–9.

[71] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. CVPR*, Jun. 2009, pp. 248–255.

[72] F. Mahmood et al., "Deep adversarial training for multi-organ nuclei segmentation in histopathology images," *IEEE Trans. Med. Imag.*, vol. 39, no. 11, pp. 3257–3267, Nov. 2020.

[73] A. Kirillov, K. He, R. Girshick, C. Rother, and P. Dollár, "Panoptic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 9404–9413.

[74] B. Alberts et al., *Essential Cell Biology*. New York, NY, USA: Garland Science, 2015.

[75] S. Javed, A. Mahmood, J. Dias, N. Werghi, and N. Rajpoot, "Spatially constrained context-aware hierarchical deep correlation filters for nucleus detection in histology images," *Med. Image Anal.*, vol. 72, Aug. 2021, Art. no. 102104.