Hierarchical Graph Pathomic Network for Progression Free Survival Prediction

Zichen Wang[1,2], Jiayun Li[1,2], Zhufeng Pan[1,3], Wenyuan Li[1,4], Anthony Sisk[1,5], Huihui Ye[1,5], William Speier[1,2,6], and Corey W. Arnold[1,2,4,5,6(✉)]

[1] Computational Diagnostics Lab, UCLA, Los Angeles, USA
cwarnold@ucla.edu
[2] The Department of Bioengineering, UCLA, Los Angeles, USA
[3] The Department of Computer Science, UCLA, Los Angeles, USA
[4] The Department of Electrical and Computer Engineering, UCLA, Los Angeles, USA
[5] The Department of Pathology and Laboratory Medicine, UCLA, Los Angeles, USA
[6] The Department of Radiological Sciences, UCLA, Los Angeles, USA

Abstract. High resolution histology images contain information related to disease prognosis. However, survival prediction based on current clinical grading systems, which rely heavily on a pathologist's histological assessment, has significant limitations due to the heterogeneity and complexity of tissue phenotypes. To address these challenges, we propose a deep learning framework that leverages hierarchical graph-based representations to enable more precise prediction of progression-free survival in prostate cancer patients. Unlike conventional approaches that analyze patch-based or cell-based pathomic features alone without considering their spatial connectivity, we explore multi-scale topological structures of whole slide images in an integrative context. Extensive experiments have demonstrated the effectiveness of our model for better progression prediction.

Keywords: Progression free survival · Graph convolutional neural network · Self-supervised learning · Hierarchical graph representations

1 Introduction

Prostate cancer (PCa) is the most common and second deadliest non-skin cancer in American men [1]. Prostate cancer progression following surgery is measured through changes in histological and biochemical measures. The ability to predict progression-free survival (PFS) could assist physicians in selecting treatment options for patients, with patients at a higher risk for progression receiving more aggressive treatment. The Gleason grading system is the current best method for PCa diagnosis and is a critical component in clinical survival assessment and treatment planning [2]. Gleason grades are assigned by pathologists

Z. Wang and J. Li—Contributed equally to this work.

Electronic supplementary material The online version of this chapter (https://doi.org/10.1007/978-3-030-87237-3_22) contains supplementary material, which is available to authorized users.

M. de Bruijne et al. (Eds.): MICCAI 2021, LNCS 12908, pp. 227–237, 2021.
https://doi.org/10.1007/978-3-030-87237-3_22

based on visual interpretation of tumor architecture. This process suffers from inter-observer variability and is constrained by its categorization system, which cannot fully capture the disease's continuous feature spectrum. To address this issue, previous machine learning methods on survival prediction focus mainly on extracting features from manually selected regions of interest (ROIs) and use pre-defined aggregation methods to generate slide-level embeddings [3–5]. However, there are two major limitations with current approaches. First, current methods ignore spatial connectivity of patches and thus fail to learn global topological representations of WSIs and may not account for tumor heterogeneity. Second, ROI-based methods often require annotations from human experts, which are expensive and can be infeasible to collect.

Deep learning methods have opened a new paradigm for automated whole slide image (WSI) analysis. However, due to image sizes, data availability, and hardware limitations, patch sampling of WSIs is necessary. To address the above challenges, graph convolutional network (GCN)-based approaches model WSIs as graph structured data, which combine global context and local connectivity with parameter efficiency. Many recent studies can be categorized into patch-based and cell-based approaches. Patch-based methods [6–9] focus on learning global topological representations of WSIs. For example, Li et al. [7] proposed a graph model for survival prediction by rendering the optimal graph representations of WSIs. Graph nodes were selected patches and node features were initialized by a VGG-16 network. Graph edges were constructed by thresholding the Euclidean distances between node features. Cell-based methods [10–12] aim to construct feature maps of patches by exploring the topological structure of cells and the micro-environment. As opposed to patch-based methods, cell-based methods emphasize the biological significance of WSIs. For example, Chen et al. [12] proposed a multimodal fusion model for cancer survival prediction. They applied GCNs as a complementary method to convolutional neural networks (CNNs) for feature extraction. They used segmented nuclei as nodes and initialized the node attributes with handcrafted features. The edges between two nodes were thresholded by a fixed spatial distance while the maximum degree of each node was set to k corresponding to its k-nearest neighbors.

Existing GCN-based methods for WSI analysis are limited in two aspects. First, patch-based methods rely on pretrained deep learning models or handcrafted features, which ignore the informative signals from cell graphs. Second, cell-based methods are inefficient in modeling WSIs with large numbers of cells, and thus typically require laborious patch-level labeling. We propose to unify patch-based and cell-based methods with a hierarchical GCN framework, balancing efficiency and granularity. We hypothesize that analyzing WSIs through multi-scale graph representations, from fine to coarse grained, will outperform current approaches for PFS prediction using WSIs. The main contributions of our paper are summarized as follows:

– To the best of our knowledge, we propose the first hierarchical GCN framework, termed graph pathomic network, for cancer survival prediction using

WSIs, which models nuclei-level and patch-level graph structures in an integrative context.
– We propose an efficient self-supervised learning method to pretrain our graph pathomic network, yielding improved performance over trained-from-scratch counterparts.
– We develop a deep learning system for PCa PFS prediction that does not rely on manually selected ROIs and patch-level labels.

2 Method

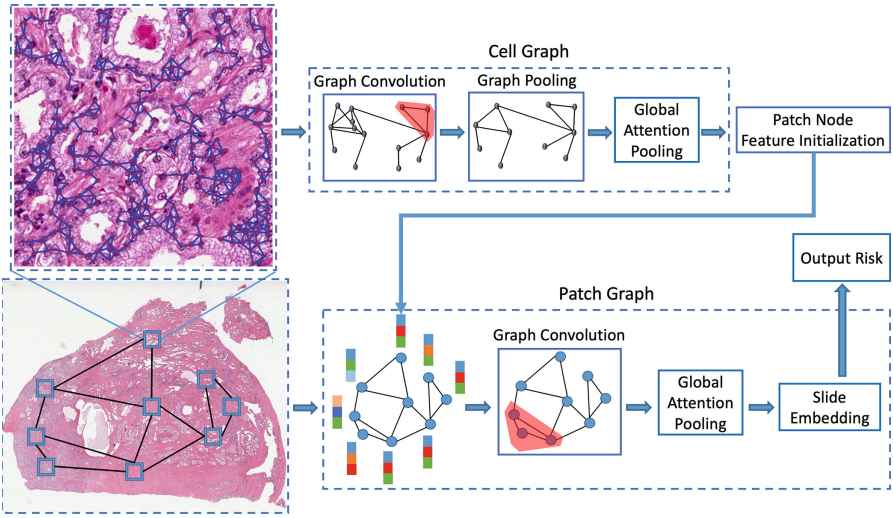

Fig. 1. Model architecture of the hierarchical graph pathomic network. Given a WSI, we sample patches using the tumor detection model to construct a patch graph, each node of which is represented by its corresponding cell graph. Our model learns hierarchical graph representations of WSIs from nuclei-level to patch-level.

2.1 Tumor Region Detection

Unlike previous studies, which rely on manually selected ROIs [3–5], we utilize a patch-level CNN trained on weakly-labeled slides to sample cancerous patches. Specifically, we formulate the problem of training a tumor detection model with slide-level labels as a multiple instance learning (MIL) task [13]. According to the basic assumption of the MIL, top patches within a malignant slide will have probabilities closer to 1, while patches from a benign slide should have probabilities closer to 0 [13]. The patches with the highest predicted probabilities in each slide inherit the slide-level label. The patch-level model is further optimized with these pseudo patch-level labels. The process is iterated until convergence. The model is then finetuned on a small dataset with coarse contour annotations.

2.2 Graph Pathomic Network

Nuclei Segmentation and Cell Feature Extraction. Accurate nuclei instance segmentation is important in defining reliable cell features that would be indicative of cancer progression. We train a Mask R-CNN [14] using the dataset from [15] to segment nuclei. After obtaining nuclei masks, we define an eight-pixel width of the ring-like neighborhood region around each nucleus as its cytoplasm area. We then extract nuclear morphometry features and imaging features (including intensity, gradient and Haralick features) for nuclear and cytoplasm regions respectively, resulting in 108-dimensional feature descriptors for each cell.

Hierarchical Graph Representation of WSIs

- **Cell Graph.** To learn complex relationships between cells for patch representation, we construct cell graphs by defining each cell as a node and an edge as the potential interaction between two cells. The intuition is that cells that are closer to each other are more likely to interact. We use the k-nearest neighbors algorithm to construct adjacency matrices of cell graphs. Specifically, we assign an edge between two nuclei if they are within a fixed spatial distance from each other. In addition, the maximum neighborhood number of each node is set to K corresponding to its k-nearest neighbors. We use K=5 to model topological information of cell graph structures, and 60 pixels at 40x resolution to set the largest distance.
- **Patch Graph.** For each WSI, we define a patch graph to aggregate patch-level features and extract topological properties of the WSI. Given a set of selected patches by the tumor region detection model, we define each patch as a node in a patch graph, and initialize its features with cell graph representations. Similar to cell graph construction, we use the k-nearest neighbors algorithm (K = 3) to construct edge sets based on spatial distance between two patches.

Graph Convolutional Network. We use the GCN with attention, also referred as the graph attention network (GAT) [16] to perform graph convolution operations to extract and propagate features along edges of the graph. Specifically, a graph $\mathcal{G} \in \mathbb{G}$ can be denoted by $\mathcal{G}(\mathcal{V}, \mathcal{E})$ consisting of a vertex set $\mathcal{V} = \{v_i\}_{i=1}^{N_\mathcal{V}}$ and edge set $\mathcal{E} = \{e_j\}_{j=1}^{N_\mathcal{E}}$, $\vec{h}_{v_i} \in \mathbb{R}^N$ is the node feature of vertex v_i. A general GCN layer can be defined as follows:

$$g_\mathcal{E}(v_i) = \text{ReLU}\left(W \cdot \text{Mean}\left(\left\{h_{v_j} | v_j \in \mathcal{N}_{v_i}^\mathcal{E} \cup \{v_i\}\right\}\right)\right). \tag{1}$$

where $W \in \mathbb{R}^{M \times N}$ is a learnable matrix transforming N-dimensional features to M-dimensional features, the Mean is an element-wise mean-pooling operation. $\mathcal{N}_{v_i}^\mathcal{E}$ is the neighborhood of the node v_i connected by $\mathcal{E}$ in $\mathcal{G}$. GAT uses an attention mechanism on node features to construct the weighting kernel as $W_{v_i,v_j} = \alpha_{v_i,v_j} W$. The weighting coefficients can be expressed as:

$$\alpha_{v_i,v_j} = \frac{\exp\left(\rho\left(\vec{a}^T\left[W\vec{h}_{v_i}\|W\vec{h}_{v_j}\right]\right)\right)}{\sum_{v_k\in\mathcal{N}_{v_i}^{\varepsilon}}\exp\left(\rho\left(\vec{a}^T\left[W\vec{h}_{v_i}\|W\vec{h}_{v_k}\right]\right)\right)} \tag{2}$$

where T represents transposition and $\|$ is the concatenation operation.

Graph Pooling. To coarsen the graph and generate high-level representation, self-attention graph pooling [17] based on both graph features and topology are used to reduce the number of nodes with a fixed pooling ratio k. After each graph pooling layer, the top k% nodes are selected based on their attention scores, and the remaining nodes are removed from the graph.

Model Architecture. Figure 1 shows an overview of our proposed model. In order to model cell-level and patch-level graph structure of WSIs in an integrative context, we build a multi-scale GCN framework using hierarchical graph representations of WSIs. Our model consists of two consecutive steps. In the first step, our model operates directly on cell graphs of all selected patches from the same slide to generate node representations for a patch graph. We use three model layers to update node features and the topological structure of cell graphs. Each layer consists of a GAT and graph pooling operations. The graph embedding $\mathbf{r_i}$ of a cell graph is obtained by applying a global soft attention-based aggregation operation [18] on updated node features $\mathbf{x_n}$, which can be expressed as:

$$\mathbf{r}_i = \sum_{n=1}^{N_i}\operatorname{softmax}\left(h_{\text{gate}}\left(\mathbf{x}_n\right)\right)\odot\mathbf{x}_n \tag{3}$$

where h_{gate} is a neural network that computes attention scores by mapping node feature vectors to scalars. In the second step, we apply two GAT layers to further update the patch graph, which is initialized by features from cell graphs. Finally, we apply the same global soft attention-base aggregation operation to generate slide embeddings for PFS prediction.

2.3 Model Training for Progression Free Survival Prediction

We utilize the negative logarithm of Cox partial likelihood proposed in the DeepSurv model [19] as the loss function for model optimization. Given the event e_i is observed for patient i at T_i, the Cox loss can be computed by Eq. 4.

$$l(\beta) = -\sum_{i:e_i=1}\left(h_\theta(\boldsymbol{z}) - log\sum_{j\in R_{(T_i)}}\exp(h_\theta(\boldsymbol{z}))\right) \tag{4}$$

where $\boldsymbol{z}$ is the slide embedding of a patient, $j \in R_{(T_i)}$ represents a set of patients with events that have not occurred at T_i. $h(\cdot)_\theta$ is a non-linear function (*e.g.,* a multi-layer perceptron).

2.4 Momentum Contrast for Unsupervised Pre-training of Cell Graph

Training an accurate GCN model often requires a large collection of labeled data and expressive features. However, we have limited patient-level labels in this study. Several recent works have demonstrated that pre-trained GCN models can benefit downstream applications by providing useful features or parameter initialization [20,21]. We generalize the Momentum Contrastive (MoCo) model [22] to graph data for pretraining the cell graph module of our proposed framework.

The main assumption for the MoCo model is that features from augmented versions of the same input should be more similar than features from different inputs. Specifically, input data can be projected into feature vectors k_i by a key encoding function $f_k(\cdot)$, which can be considered as keys in a dataset. Given the encoded representation $\boldsymbol{q_i}$ of a query data generated by the query encoding function $f_q(\cdot)$, there exists one matched key, which has the largest similarity value to $\boldsymbol{q_i}$. The similarity between a pair of feature vectors can be measured using the cosine distance: $\mathrm{sim}(\boldsymbol{q_i}, \boldsymbol{k_i}) = \frac{\boldsymbol{q_i}\boldsymbol{k_i}}{\|\boldsymbol{q_i}\|\|\boldsymbol{k_i}\|}$. The contrastive loss is defined by Eq. 5.

$$\mathcal{L}_q = -\log \frac{\exp(\mathrm{sim}(\boldsymbol{q_i}\boldsymbol{k_i})/\gamma)}{\exp(\mathrm{sim}(\boldsymbol{q_i}\boldsymbol{k_i})/\gamma) + \sum_{j\neq i}^{N} \exp(\mathrm{sim}(\boldsymbol{q_i}\boldsymbol{k_j})/\gamma)} \tag{5}$$

where γ is the temperature parameter. $\boldsymbol{q_i}$ and $\boldsymbol{k_i}$ are feature vectors from different views of the same data i. Importantly, MoCo improves conventional contrastive learning by building a dynamic dictionary with a queue and a moving-averaged encoder. This enables building a large and consistent dictionary on-the-fly that facilitates contrastive unsupervised learning. Specifically, we define two augmentation methods to generate different views of the same cell graph. (1) Node dropping: we randomly drop cell nodes with a fixed percent number (5%) from the original graph. (2) Graph structure perturbation: we randomly set hyperparameters for cell graph construction from a fixed range of values, which also reduces model sensitivity to parameters. In this study, hyperparameters include maximum neighborhood number (5–7) and the largest distance between connected nodes (55–70 px).

3 Experiment

3.1 Dataset and Preprocessing

Three datasets are used in this experiment. (1) A UCLA prostate biopsy dataset containing 20,229 slides from prostate needle biopsies from 830 patients [13]. (2) A Cedars-Sinai dataset consisting of 30 slides from prostatectomies of 30 patients, which were annotated with coarse contour annotations [23]. (3) The PFS prediction is performed on the publicly available TCGA-PRAD dataset [24]

using patches of size 512×512 at 40x magnification(0.25 um per pixel). Datasets (1) and (2) are used to train the tumor detection model. For dataset (3), clinical data and H&E stained diagnostic slides were retrieved using the Genomic Data Commons (GDC) data portal. Additional follow-up data are obtained from the TCGA Pan-Cancer Clinical Data Resource (TCGA-CDR) [24]. The recommended progression-free interval (PFI) is used as the clinical endpoint [24]. 399 cases with available diagnostic slides are included in this study. Five-fold cross validation stratified by events is utilized for model training and validation. Within each fold, we further split the training data into 80% for training and 20% for validation. Model performance is evaluated on testing data for each fold to avoid overfitting.

3.2 Methods Comparison

To evaluate the effectiveness of our hierarchical graph pathomic network, we performed experiments on different types of features, aggregation methods, and linear Cox models.

Clinical Features. We utilized clinical variables including age, PSA value, Gleason grade group (GG), and pathologic T stage to fit a baseline Cox model.

Linear Cox Model. We compared our model with linear Cox models, including Lasso-Cox (*i.e.,*, with $\mathcal{L}_1$ penalty) and ElasticNet-Cox (*i.e.,* with $\mathcal{L}_2$ penalty). Different types of features as described in the following sections were used as covariates for linear Cox models.

Attention MIL-Based Aggregation. We compared the GCN-based aggregation with the attention MIL method, which demonstrated promising results in GG prediction [13]. Specifically, a trainable attention module was incorporated to summarize patch-level features into slide embeddings.

Texture-Based Features. We extracted 90 first-order statistics and texture-based features with the Pyradiomics package [25]. Texture-based features were utilized as predictors in *ElasticNet-Cox-texture* and *Lasso-Cox-texture* experiments.

Self-supervised CNN Features. We adopted the MoCo model with ResNet50 as the backbone [22] to learn CNN features from patches. The trained query encoder of the MoCo model was used as the feature extractor to generate a 128×1 feature vector for each patch. Self-supervised learning features were utilized as covariates in the *Lasso-Cox-deep* and *ElasticNet-Cox-deep* experiments. *Deep-Att-MIL* denotes the deep survival model that leverages self-supervised CNN features and attention MIL-based aggregation. *Deep-GCN* denotes the variant of our proposed method, which includes only the patch graph module and uses self-supervised CNN features as initial node embeddings.

Handcrafted Cell Graph Features. To validate the effectiveness of the cell graph, we extract pre-defined cell graph features for the ablation study. We compute global features of the nuclei with the given centroids based on the partitioning of the space into Voronoi cells and on the induced graph structure. This type of feature only captures the topological structure of cell graph.

3.3 Model Training and Evaluation

We selected 200 patches from each slide for analysis. We used a learning rate of 0.001 and a batch size of 64 for Deep-Att-MIL model training. The learning rate and batch size for Deep-Att-MIL-clinical were set at 0.001 and 12, respectively. All GCN-based models were trained with a batch size of 16 and a learning rate of 0.0005. Adam optimizer was used in all experiments. PFS analysis models were evaluated with the concordance index (c-index), which was computed with the lifelines package [26]. The c-index measures the effectiveness of predicted risk scores on ranking survival times, with 1 indicating the perfect concordance and 0.5 representing results from random predictions. For cases with multiple slides, average hazards were utilized. All models were implemented with PyTorch, trained and validated with one Tesla V100 GPU.

Table 1. Concordance index of models on predicting progression free survival for prostate cancer patients following radical prostatectomy.

Models	c-index ↑
Clinical features	0.7254 ± 0.042
Lasso-Cox-texture	0.6818 ± 0.067
Lasso-Cox-graph	0.6857 ± 0.092
Lasso-Cox-deep	0.7046 ± 0.111
ElasticNet-Cox-texture	0.6671 ± 0.052
ElasticNet-Cox-graph	0.6892 ± 0.084
ElasticNet-Cox-deep	0.7254 ± 0.101
Deep-Att-MIL	0.7420 ± 0.089
Deep-GCN	0.7441 ± 0.123
Graph pathomic network	0.7539 ± 0.103
Pre-trained graph pathomic network	**0.7636 ± 0.091**
Deep-Att-MIL + clinical	0.7586 ± 0.076
Deep-GCN + clinical	0.7743 ± 0.122
Graph pathomic network + clinical	0.7831 ± 0.105
Pre-trained graph pathomic network + clinical	**0.7934 ± 0.082**

4 Results

Average c-index and 95% confidence interval for model experiments are shown in the Table 1. The baseline Cox model with clinical features obtained an average c-index of 72.54%. The pre-trained graph pathomic network that combined clinical variables achieved the highest average c-index of 79.34%, which was around 7% higher than the baseline model with only clinical features. The spatial distribution of nuclei is important for prostate progression estimation. As shown in the Table 1, models with handcrafted graph-based features showed better performances than those with texture-based features. Models with deep learning representations demonstrated superior performance compared with handcrafted features. The GCN based aggregation method showed better results in predicting progression compared to Att-MIL aggregation. The graph pathomic network outperformed all baseline models, demonstrating the effectiveness of learning features from hierarchical graphs. Graph MoCo pre-training brought around 1% c-index improvement than trained-from-scratch counterparts. We also observed that pre-trained graph pathomic network converged faster during model training.

In summary, we developed a deep learning system with novel hierarchical graph-based representations for PFS prediction of PCa patients. By building hierarchical graph representations, our proposed model shows promising results on predicting PFS. In addition, combining clinical factors and pathomic features is superior to the baseline model with clinical features only, suggesting the potential of leveraging quantitative pathomic features for better progression prediction. However, we only evaluated our model on the TCGA-PRAD dataset. Validation on large prospective datasets is needed to further validate pathomic features. Reinforcement learning may also be investigated in future work to enable adaptive patch selection and end-to-end training.

References

1. Siegel, R.L., Miller, K.D., Jemal, A.: Cancer statistics, 2019. CA Cancer J. Clin. **69**(1), 7–34 (2019)
2. Epstein, J.I., et al.: A contemporary prostate cancer grading system: a validated alternative to the Gleason score. Euro. Urol. **69**(3), 428–435 (2016)
3. Chandramouli, S., et al.: Computer extracted features from initial H&E tissue biopsies predict disease progression for prostate cancer patients on active surveillance. Cancers **12**(9), 2708 (2020)
4. Leo, P., et al.: Computerized histomorphometric features of glandular architecture predict risk of biochemical recurrence following radical prostatectomy: a multisite study (2019)
5. Cheng, L., et al.: Nuclear shape and orientation features from H&E images predict survival in early-stage estrogen receptor-positive breast cancers. Lab. Invest. **98**(11), 1438–1448 (2018)

6. Zhao, Y., et al.: Predicting lymph node metastasis using histopathological images based on multiple instance learning with deep graph convolution. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4837–4846 (2020)
7. Li, R., Yao, J., Zhu, X., Li, Y., Huang, J.: Graph CNN for survival analysis on whole slide pathological images. In: Frangi, A.F., Schnabel, J.A., Davatzikos, C., Alberola-López, C., Fichtinger, G. (eds.) MICCAI 2018. LNCS, vol. 11071, pp. 174–182. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-00934-2_20
8. Adnan, M., Kalra, S., Tizhoosh, H.R.: Representation learning of histopathology images using graph neural networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pp. 988–989 (2020)
9. Ding, K., Liu, Q., Lee, E., Zhou, M., Lu, A., Zhang, S.: Feature-enhanced graph networks for genetic mutational prediction using histopathological images in colon cancer. In: Martel, A.L., et al. (eds.) MICCAI 2020. LNCS, vol. 12262, pp. 294–304. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-59713-9_29
10. Zhou, Y., Graham, S., Koohbanani, N.A., Shaban, M., Heng, P.-H., Rajpoot, N.: CGC-net: cell graph convolutional network for grading of colorectal cancer histology images. In: Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (2019)
11. Wang, J., Chen, R.J., Lu, M.Y., Baras, A., Mahmood, F.: Weakly supervised prostate TMA classification via graph convolutional networks. In: 2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI), pp. 239–243. IEEE (2020)
12. Chen, R.J., et al.: Pathomic fusion: an integrated framework for fusing histopathology and genomic features for cancer diagnosis and prognosis. IEEE Trans. Med. Imaging (2020)
13. Li, J., et al.: A multi-resolution model for histopathology image classification and localization with multiple instance learning. In: Computers in Biology and Medicine, p. 104253 (2021)
14. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2961–2969 (2017)
15. Kumar, N., et al.: A multi-organ nucleus segmentation challenge. IEEE Trans. Med. Imaging **39**(5), 1380–1391 (2019)
16. Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., Bengio, Y.: Graph attention networks. arXiv preprint arXiv:1710.10903 (2017)
17. Lee, J., Lee, I., Kang, J.: Self-attention graph pooling. In: International Conference on Machine Learning, pp. 3734–3743. PMLR (2019)
18. Li, Y., Tarlow, D., Brockschmidt, M., Zemel, R.: Gated graph sequence neural networks. arXiv preprint arXiv:1511.05493 (2015)
19. Katzman, J.L., Shaham, U., Cloninger, A., Bates, J., Jiang, T., Kluger, Y.: Deep survival: a deep cox proportional hazards network. Stat **1050**(2) (2016)
20. Hu, W., et al.: Strategies for pre-training graph neural networks. arXiv preprint arXiv:1905.12265 (2019)
21. Qiu, J., et al.: GCC: graph contrastive coding for graph neural network pre-training. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 1150–1160 (2020)
22. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9729–9738 (2020)
23. Ing, N., et al.: Semantic segmentation for prostate cancer grading by convolutional neural networks. In: Medical Imaging 2018: Digital Pathology, vol. 10581, pp. 105811B. International Society for Optics and Photonics (2018)

24. Liu, J., et al.: An integrated TCGA pan-cancer clinical data resource to drive high-quality survival outcome analytics. Cell **173**(2), 400–416 (2018)
25. Van Griethuysen, J.J.M., et al.: Computational radiomics system to decode the radiographic phenotype. Cancer Res. **77**(21), e104–e107 (2017)
26. Davidson-Pilon, C., et al.: Camdavidsonpilon/lifelines: v0. 24.15. Zenodo (2020)