Generating Hypergraph-Based High-Order Representations of Whole-Slide Histopathological Images for Survival Prediction

Donglin Di, Changqing Zou, *Member, IEEE*, Yifan Feng, Haiyan Zhou, Rongrong Ji, *Senior Member, IEEE*, Qionghai Dai, *Senior Member, IEEE*, and Yue Gao, *Senior Member, IEEE*

Abstract—Patient survival prediction based on gigapixel whole-slide histopathological images (WSIs) has become increasingly prevalent in recent years. A key challenge of this task is achieving an informative survival-specific global representation from those WSIs with highly complicated data correlation. This article proposes a multi-hypergraph based learning framework, called "HGSurvNet," to tackle this challenge. HGSurvNet achieves an effective high-order global representation of WSIs via multilateral correlation modeling in multiple spaces and a general hypergraph convolution network. It has the ability to alleviate over-fitting issues caused by the lack of training data by using a new convolution structure called hypergraph max-mask convolution. Extensive validation experiments were conducted on three widely-used carcinoma datasets: Lung Squamous Cell Carcinoma (LUSC), Glioblastoma Multiforme (GBM), and National Lung Screening Trial (NLST). Quantitative analysis demonstrated that the proposed method consistently outperforms state-of-the-art methods, coupled with the Bayesian Concordance Readjust loss. We also demonstrate the individual effectiveness of each module of the proposed framework and its application potential for pathology diagnosis and reporting empowered by its interpretability potential.

Index Terms—High-Order representation, hypergraph learning, survival prediction, whole slide image

1 Introduction

Digital pathology has developed considerably over the past few decades as it has become possible to digitize slides into "whole slide images" (WSIs) [1], [2]. These heavy gigapixel images contain all the information required to diagnose lesions [3]. The task of survival prediction, which aims to model the life duration of a patient based on WSIs, has attracted more and more attention in recent years [4], [5], [6], [7], because it has significant application potential in aiding the effort of pathologists through an objective analysis. The key challenge in this task is how to extract the informative and effective survival-specific representation reflecting the patient's survival status for the model to make an accurate prediction. However, unlike those of regular natural images such as that in ImageNet [8], the WSI data may be extremely large, i.e., an image may have billions of pixels and the data correlation is highly complicated. This scenario makes existing well-performing and sophisticated models [9], [10] which are designed for analyzing natural images of a much smaller size like $256px \times 256px$ hardly applicable for those gigapixel histopathological images.

To make those WSI data interpretable for an AI model, recent methods [4], [7], [11] follow a pipeline as such: first sample a number of patches with an affordable size (e.g., 256×256) from each WSI, and then stack those patches and feed them into a CNN based feature extractor (e.g., VGG [12]) to generate a global representation by treating each patch as a channel as shown in Fig. 1b. After that, apply a regression model to the global feature to predict survival scores. The representation learning ability of these methods is mainly limited by the fact that the structure of the whole histopathological image is broken up by patch sampling. DeepConvSurv [13] is the first WSI compliant CNN-based survival prediction model. To model complex correlation structures among patches, DeepGraphSurv [5] applies a graph convolutional neural network to low-level patch features generated from a CNN based feature extractor. However, squeezing complex correlation into pairwise correlation could inevitably lead to the loss of the information [14] believed to be valuable for

- *Donglin Di, Yifan Feng, Qionghai Dai, and Yue Gao are with BNRist, KLISS, School of Software, BLBCI, THUIBCS, Tsinghua University, Beijing 100084, China. E-mail: {donglin.ddl, evanfeng97}@gmail.com, {qionghaidai, gaoyue}@tsinghua.edu.cn.*
- *Changqing Zou is with the State Key Lab of CAD&CG, Zhejiang Lab, Zhejiang University, Hangzhou 310027, China. E-mail: aaronzou1125@gmail.com.*
- *Haiyan Zhou is with the Department of Pathology, Xiangya hospital, Central South University, Changsha, Hunan 410017, China. E-mail: yanhaizhou78@163.com.*
- *Rongrong Ji is with Media Analytics and Computing Laboratory, Department of Artificial Intelligence, School of Informatics, Institute of Artificial Intelligence, Xiamen University, Xiamen 361005, China, and also with Peng Cheng Laboratory, Shenzhen 518066, China. E-mail: rrji@xmu.edu.cn.*

Manuscript received 13 December 2020; revised 12 May 2022; accepted 18 September 2022. Date of publication 26 September 2022; date of current version 3 April 2023.
This work was supported in part by the National Natural Science Funds of China under Grants 62088102 and 62021002, in part by the Open Research Projects of Zhejiang Lab under Grant 2021KG0AB05, and in part by Beijing Natural Science Foundation under Grant 4222025.
(Corresponding author: Yue Gao.)
Recommended for acceptance by D. Samaras.
Digital Object Identifier no. 10.1109/TPAMI.2022.3209652

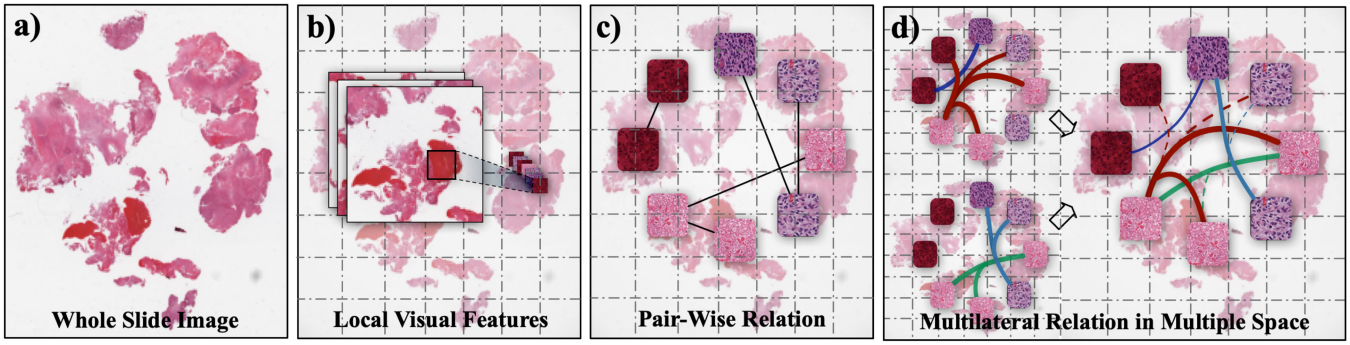

Fig. 1. Existing survival prediction methods predict the life duration of a patient by using either local patch-level visual features of WSI data (a) extracted with convolution networks [4] (b) or their embeddings obtained by feature aggregation through pairwise relations [5] (c). Our approach HGSurvNet (d) instead performs the prediction by explicitly modeling a higher-order global representation for WSIs. This is achieved by feature aggregation through multilateral relations in multiple spaces.

survival prediction. As shown in Fig. 1c, data correlation in this method is constructed on pair-wise patches, making it relatively weak to exploit the pathology related data correlation, which is believed to be a high-order information [11], [15].

This paper addresses the challenge by generating an effective high-order survival-specific data correlation representation with a multi-hypergraph based framework called "HGSurvNet," to more powerfully model higher-order correlations of the sampled WSI patches than previous methods. Our framework is mainly empowered by two technical contributions. The first technical contribution of HGSurvNet is a multi-hypergraph learning-based framework for high-order correlation modeling of the WSI data. In recent years, hypergraph learning has shown significant advantages in modeling complex relations in various data such as 3D point cloud [16], hyperspectral image [17], multi-modality social media data [18], signaling pathways of cells [19]. To achieve an effective global representation for WSI data, in this paper we explicitly model the multilateral data correlation with hypergraphs in multiple spaces and promote the modeling into a high-order level using a multi-hypergraph learning framework. Specifically, we construct the domain-specific sub-hypergraphs from WSI patches in two typical spaces: latent feature space and image space. This leads to two types of sub-hypergraphs, phenotype-wise sub-hypergraph and topology-wise sub-hypergraph as illustrated in Fig. 1d, which can explore the data correlations related to both visual appearance and spatial structural information in the same multi-hypergraph learning framework simultaneously. To cooperate with the multilateral data correlation modeling, we also design a powerful high-order feature fusion module which aggregates both node and hyperedge level high-order features to form an effective global representation for survival prediction.

The second technical contribution of HGSurvNet is a general hypergraph network called hypergraph max-mask convolution (HGMConv for short). This network built on general spectral hypergraph convolutional layers [20], [21] can alleviate the over-fitting issue caused by a limited amount of training data and help the model perform better on testing data. Our study presented in the experiments, finds the proposed hypergraph max-mask convolution can improve both robustness and accuracy of survival prediction. Considering the fact that the labeling process of WSI data requires specialized diagnosis training and is costly [1], [2], [13], and medical privacy is a major concern for many patients, the hypergraph max-mask convolution, therefore, is of great significance when applying hypergraph learning to the WSI data.

Based on the two contributions mentioned above, we design a three-stage architecture (i.e., Fig. 2), where the core multi-hypergraph learning network and survival status regression network can be trained in an end-to-end manner. With this design, the proposed method consistently outperforms state-of-the-art methods (DeepConvSurv [13], WSISA [4], GCN [22], DeepGraphSurv [5], DeepMISL [6], PatchGCN [23]) by a large margin on three datasets for the task of survival prediction using WSIs. Specifically, with the same loss function (i.e., the negative Cox log partial likelihood loss function), it improves the prediction accuracy by up to 8.6% on two lung cancer datasets (i.e., LUSC [24] and NLST [25]) and 9.1% on a brain carcinoma dataset (GBM [24]).

Moreover, through the interpretability study, we have found that the proposed method is helpful for pathology diagnosis and reporting, because of its capacity for pathology region location, distribution pattern and intensity visualization, as well as phenotype identification and cell microenvironment analysis.

The remainder of this paper is organized as follows. Section 2 summarizes previous works. Section 3 details each component module of HGSurvNet. Section 4 comprehensively evaluates HGSurvNet. The last section concludes the paper and discusses the future work.

2 Related Work

2.1 Survival Analysis

First, traditional statistical methods for survival analysis, estimating the survival/hazard functions, are most commonly

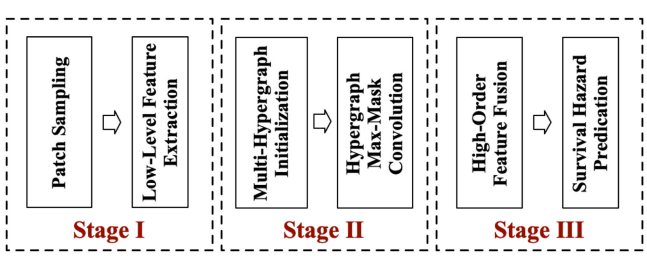

Fig. 2. Overall framework of our proposed HGSurvNet, containing three pipeline stages. .

divided into three branches, i.e., non-parametric, semi-parametric, and parametric. In non-parametric methods, the widely adopted algorithms are Kaplan-Meier (KM) [26], [27], Nelson-Aalen estimator (NA) [28] or Life-Table (LT) [29], to obtain the survival function by the empirical estimating. For the semi-parametric category, Cox model [30] is the most commonly used regression approach for survival data, by building the proportional hazards assumption and partial likelihood for parameter estimation. Based on Cox, there are several popular models belonging to the semi-parametric branch, such as Lasso-Cox [31], Ridge-Cox [32], En-Cox [33], [34] and so on. Parametric methods are more efficient and accurate for estimating the survival time, such as linear regression [35], exponential distribution, Weibull distribution, logistic and log-logistic distribution, normal and log-normal distribution *etc.*

Besides statistical methods, a number of machine learning models have been proposed in the past years. Survival trees [36], Bayesian methods [37], [38], artificial neural networks [37], [39], support vector machines (SVM) [40], [41], [42], random survival forests (RSF) [43], boosting concordance index (BoostCI) [44] *etc.*perform more effectively and achieve better performance when picking property hyper-parameters of models.

Recently, deep-learning based models have made a dramatic leap over these two branches of classical approaches listed above. Convolutional Neural Networks (CNN), as the trunk model, have been extensively explored and seen a lot of successful applications recently. The DeepConvSurv [13] proposes a deep convolutional survival model which relieves the demand for large volume samples by augmenting image patches. A major limitation of this model is that the model training needs patch-wise samples. WSISA [4] therefore tries to address this limitation by performing survival prediction on whole slide pathological images in an end-to-end manner. However, approaches based on CNN [4], [7], [13] have an inherent limitation in modeling global topological information, which appears to be vital for analyzing gigapixel WSIs. This means that theoretically, the survival prediction model cannot perform very well simply based on the constrained localized receptive field.

Graph Neural Networks (GNN) [22], [45], [46] have become a widely used technique because of their ability to analyze graph structural data. Recently, DeepGraphSurv [5] utilized spectral graph convolution operators to integrate both local patch features and global topological features, learnt on patch pairs, simultaneously. Global topological information and the correlation between patches can be extremely complex and are difficult to model through the pair-wise graph structures employed in DeepGraphSurv. The proposed method differs from DeepGraphSurv [5] mainly in that: (1) it employs a multi-hypergraph based framework to model higher-order correlations rather than a graph based framework modeling pair-wise correlations of the sampled patches, and (2) its representation is built on both node and hyperedge levels of features rather than only the node level features used in DeepGraphSurv.

Besides these aforementioned single-modality models, there are also some multimodal models which learn pathological properties from different modality data including WSI and genetic data [47], [48], [49], [50]. These methods are theoretically more effective for survival analysis because they can utilize the complementary information presented in multimodal data.

2.2 Preliminary on Hypergraph Learning

Hypergraph learning has been widely applied in many tasks such as identifying non-random structures in the structural connectivity of the cortical microcircuits [21], identifying high-order brain connectome biomarkers for disease diagnosis [51], and studying the co-relationships between functional and structural connectome data [52]. Hypergraph learning was first introduced in [53], in which each node represents one case, each hyperedge captures the correlation between each group of nodes, and the learning process is conducted on a hypergraph as a propagation process. By this method, the transductive inference on the hypergraph aims to minimize the label differences between vertices that are connected by more and stronger hyperedges. Then, the hypergraph learning is conducted as a label propagation process on the hypergraph to obtain the label projection matrix [54], or as a spectral clustering [55]. Other applications of hypergraph learning include video object segmentation [56], images ranking [57], and landmark retrieval [58]. Hypergraph learning has the advantage of modeling high-order correlation, but the reliability of different vertices on the hypergraph, also important to conduct accurate learning, has not been thoroughly investigated.

Hypergraph Neural Networks (HGNN) [20] have been designed to tackle the challenge of modeling complex high-order representations and formulating cross-modality data correlations. The hypergraph convolution operation is proposed to more effectively exploit high-order data correlation for representation learning (e.g., cell subtype appearance, cancer or tumor structure, tissue hierarchy in our task). The original HGNN model [20] mainly focuses on general tasks like image classification and visual object recognition. Instead of modeling multi-modal data, we propose to take different hyperedges as separate factors and utilize the hypergraph structure to represent each effect of them.

3 Method

HGSurvNet comprises three stages. The first stage samples dense and informative patches from the whole-slide histopathological image and extracts their low-level visual features. Gigapixel histopathological images have many pixels corresponding to healthy tissues and cells irrelevant to survival hazard; survival status prediction based on the sampled informative patches with a reasonable size makes the learning feasible. The second stage performs multi-hypergraph learning to model high-order correlation among the sampled patches. In this stage, we introduce a multi-hypergraph structure to generate the weights for the sub-hypergraphs constructed from the complex correlation of different features. This stage can perform node-hyperedge-node feature transformation which can better refine the features with the multi-hypergraph structure. The max-mask learning strategy is used to address the over-fitting issue in this stage, i.e., avoiding the situation where only dominant nodes (hyperedges) contribute to the representation learning. The third stage first forms a global representation for the patches

whose features have been refined by the multi-hypergraph learning, and then predicts the survival hazard score with a regression neural network. In this stage, the global representation is achieved by aggregating two-level features: both node- and hyperedge- level features are integrated to generate a global feature for the final regression. The three stages work together and provide an effective and robust, pathological representation for the WSI data.

Algorithm 1. Patch Sampling

1: **function** SamplePatches($\mathcal{W}$, N, $\mathcal{T}$, $\mathbf{H}$)
2: $slide \leftarrow$ OpenSlide($\mathcal{W}$) ▷ Read WSI $\mathcal{W}$.
3: $\mathcal{R} \leftarrow$ Otsu$slide$, ($\mathcal{T}$) ▷ Segment the ROI.
4: $\mathbf{P}, \mathbf{G} \leftarrow \Phi$ ▷ Initialize the patch and grid lists.
5: $\mathbf{G} \leftarrow$ TileGrids($\mathcal{R}$) ▷ $\mathbf{G} \in \mathbb{R}|_0^r, r_1, \ldots, r_k|^\top \times 4$,
6: ▷ $||\mathcal{R}||_0 = k$, includes k ROIs and corresponding spatial coordinates.
7: $\mathbb{G}_{rand} \leftarrow$ RandomPick($\mathbf{G}$) ▷ Randomly select the patches.
8: $\mathbb{G}_{topo} \leftarrow$ TopologicalPick($\mathbf{G}$)
9: $\mathbb{G} \leftarrow \mathbb{G}_{rand} \cup \mathbb{G}_{topo}$
10: $\mathcal{M}_{pig} \leftarrow$ MeanPigment($\mathbb{G}$)
11: ▷ Compute the mean pigment value.
12: **if** $\mathcal{M}_{pig} \in \mathbf{H}$ **then** ▷ Exclude patches of low pigment.
13: $\mathbf{P} \leftarrow \mathbf{P} \cup (pig, \mathcal{M}_{pig})$
14: **end if**
15: $\widehat{\mathbf{P}} \leftarrow$ Sort($\mathbf{P}$, $\mathcal{M}_*$) ▷ Sort patches by pigment value.
16: $\widetilde{\mathbf{P}} \leftarrow$ Subset$\widehat{\mathbf{P}}$, N ▷ Keep top N patches and discard others.
17: **return** $\widetilde{\mathbf{P}}$
18: **end Function**
19:
20: **function** TopologicalPick($\mathbf{G}$)
21: ▷ Input $\mathbf{G}$ include k ROI, each ROI of which includes multiple patches; output the topological patches.
22: $\mathbb{C}, \mathbb{B}^{\frac{1}{4}}, \mathbb{B}^{\frac{1}{2}}, \mathbb{B}^{\frac{3}{4}}, \mathbb{B}^1, \mathbb{G} \leftarrow \Phi$ ▷ Initialization.
23: **for** $r_i \in \mathcal{R}$ **do**
24: $grids_i \leftarrow \mathbf{G}_i$
25: $c_i, b_i^{\frac{1}{4}}, b_i^{\frac{1}{2}}, b_i^{\frac{3}{4}}, b_i^1 \leftarrow$ Pick($grids_i$)
26: $\mathbb{C}, \mathbb{B}^{\frac{1}{4}}, \mathbb{B}^{\frac{1}{2}}, \mathbb{B}^{\frac{3}{4}}, \mathbb{B}^1 \leftarrow \{\mathbb{C}, \mathbb{B}^{\frac{1}{4}}, \mathbb{B}^{\frac{1}{2}}, \mathbb{B}^{\frac{3}{4}}, \mathbb{B}^1\} \cup \{c_i, b_i^{\frac{1}{4}}, b_i^{\frac{1}{2}}, b_i^{\frac{3}{4}}, b_i^1\}$
27: **end for**
28: **return** $\mathbb{G} \leftarrow \{\mathbb{C}, \mathbb{B}^{\frac{1}{4}}, \mathbb{B}^{\frac{1}{2}}, \mathbb{B}^{\frac{3}{4}}, \mathbb{B}^1\}$
29: **end Function**

3.1 Patch Sampling and Low-Level Feature Extraction

To extract as much information related to survival hazards as possible, existing methods densely sample random candidate patches from WSIs [4], [5], [6], [11], or sample patches from diagnostically-relevant regions annotated by experts [1]. A limited amount of randomly sampled patches may discard the topological structures relevant to survival status prediction as pointed out in [5]. High-quality sampling can be obtained using the annotation from pathological experts, but it is costly. Differing from existing methods, we instead sample the topological patches in informative regions of WSIs as shown in Fig. 3. The major characteristic of our extracted patches is that it can preserve the topological information of the regions of interests. Our experiments find that combining topological patches based sampling and random sampling strategies can provide a more effective extraction of WSIs.

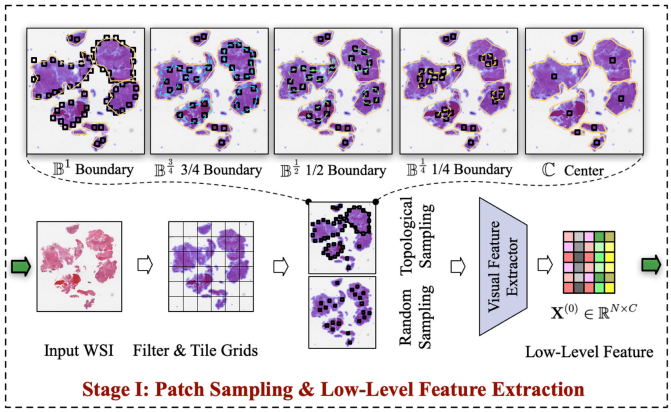

Fig. 3. Patch sampling and low-level feature extraction. .

Algorithm 1 summarizes the steps of our patch sampling. Given a WSI, we first segment out the regions of interests (ROIs for short, i.e., informative regions with tissues or cells) and discard background regions by leveraging the OTSU algorithm [59] (lines 2-5). We use the OTSU algorithm for this step because it can achieve high-quality segmentation through an adjustable gray threshold. After that, we perform two types of patch sampling. we first randomly extract patches in the ROIs and then extract the patches around the boundaries and centers of those ROIs (lines 20-28). In this step, besides of the boundary $\mathbb{B}^1$ of an ROI, we also extract patches along the "concentric sub-boundaries," which we generate by scaling down $\mathbb{B}^1$ around its center $\mathbb{C}$ by $\frac{3}{4}, \frac{1}{2}, \frac{1}{4}$ (i.e., $\mathbb{B}^{\frac{1}{4}}, \mathbb{B}^{\frac{1}{2}}, \mathbb{B}^{\frac{3}{4}}$, as visualized in the top row of Fig. 3). Lastly, we discard those patches with lower mean pigment value and keep a fixed number, N, of informative patches (lines 10-17).

After sampling topological patches, we extract the low-level visual features of each patch by employing ResNet [60] pre-trained on ImageNet [8]. Histopathology images are dominated by repetitive structures and tissues [1], which are less complex in terms of visual appearance than those natural image samples in ImageNet [8]. The pre-trained feature extractor is effective enough to extract low-level visual features needed by the subsequent multi-hypergraph learning. In this way, each patch is represented by a feature vector $\mathcal{F}_v \in \mathbb{R}^{1 \times C}$ (C is the dimension or length of the feature vector), and the first stage produces N feature vectors $\mathbf{X} \in \mathbb{R}^{N \times C}$ that represents a histopathology image.

3.2 Multi-Hypergraph Learning

In this stage, we explicitly model the complex high-order relationships among the topological WSI patches extracted in the first stage. Since different types of hypergraphs can be constructed given those patches, we formulate this processing as a problem of multi-hypergraph learning. Existing methods [4], [7], [13], explore the data correlation mainly through the visual features. Our approach provides a flexible framework that is able to model higher-order data correlation by using multiple feature information. In our implementation, given the extracted topological WSI patches, inspired by [5], we use two important types of information, i.e., phenotype (visual appearance), and topology information, for the multi-hypergraph learning. Specifically, the learning stage includes two sequential modules: multi-hypergraph initialization and hypergraph convolution consisting of

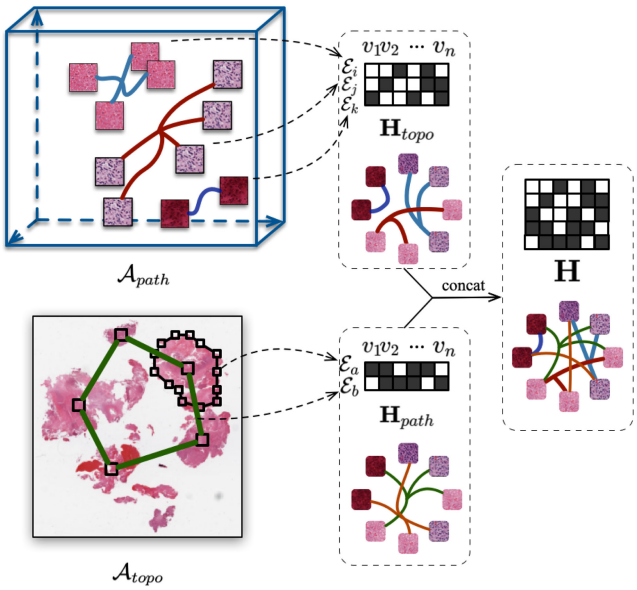

Fig. 4. Multi-hypergraph construction. Two types of hypergraph, topology-wise sub-hypergraph and phenotype-wise sub-hypergraph, are used for multi-hypergraph learning.

several hypergraph convolutional layers. The former constructs a multi-hypergraph (i.e., hyperedge groups), and the latter refines the node features as well as achieves hyperedge level features. Both node features and hyper-edge level features are further aggregated in the next stage. To alleviate the data over-fitting issue, in this stage, we do not directly use a general learning strategy for hypergraph convolution, instead, we design a max-mask learning strategy. This strategy functions similarly to the dropout strategy. It drops the contribution of the most significant hyperedges (nodes) during the feed forward/backward propagation in each layer, which essentially allows more nodes to contribute to the presentation learning. We call the hypergraph convolutional layers empowered by this max-mask training strategy as hypergraph max-mask convolutional layers. Next, we detail the multi-hypergraph initialization module and the hypergraph max-mask convolution.

3.2.1 Multi-Hypergraph Initialization

The multi-hypergraph in this step is constructed by concatenating two sub-hypergraphs (i.e., hyperedge group), as illustrated in Fig. 4. Each sub-hypergraph is constructed following a general hypergraph construction procedure. By denoting a hypergraph as $\mathcal{G} = \langle \mathcal{V}, \mathcal{E}, \mathbf{W} \rangle$, where $\mathcal{V} = [v_i], i \in (1, N)$ and $\mathcal{E} = [e_j], j \in (1, E)$, respectively, are the set of the nodes and hyperedges, $\mathbf{W} = [w_j], j \in (1, E)$ is the weight of the hyperedges. We take the extracted patches from the first stage as the nodes in a sub-hypergraph.

The initialization includes the following three major steps.

1) In the first step, we stack the N patch features $\mathcal{F}_v \in \mathbb{R}^{1\times C}$, where C denotes the length of the low level feature vector, and initialize the hypergraph signal matrix $\mathbf{X}^{(l)} \in \mathbb{R}^{N\times C}$ at the l_{th} layer.
2) In the second step, we generate hyperedges to associate a node and its "neighbors". We consider neighbor recognition of a node in both the latent feature space and the image space. In the latent feature space, two patches with similar feature vectors may be connected by a common hyperedge. In contrast, in the image space two neighboring patches on a common topological path may be connected by the same hyperedge. We name the sub-hypergraph generated in the above two spaces as phenotype-wise sub-hypergraph and topology-wise sub-hypergraph, specifically, as shown in Fig. 5. For each sub-hypergraph, we generate a hypergraph incidence matrix, i.e., $\mathbf{H}_{phe}$ for phenotype-wise sub-hypergraph, and $\mathbf{H}_{top}$ for topology-wise sub-hypergraph, respectively.
3) In the last step, we concatenate the two sub-hypergraph and form a combined hypergraph incidence matrix $\mathbf{H}$ from $\mathbf{H}_{top}$ and $\mathbf{H}_{phe}$.

Specifically, in a phenotype-wise sub-hypergraph, each node is connected with its K nearest neighbors according to the euclidean distance between the visual features of each pair of nodes, i.e., $d(x_i, x_j), i, j \in [1, N]$, shown in Eq. (1)

$$d(x_i, x_j) = \left(\sum_{a=0}^{C-1} (\mathcal{F}_{v_i}[a] - \mathcal{F}_{v_j}[a])^2 \right)^{\frac{1}{2}}, \tag{1}$$

where $\mathcal{F}_{v_i}$ and $\mathcal{F}_{v_j}$ denote the low-level visual features of node v_i and v_j, respectively. Therefore, the incidence matrix $\mathbf{H}_{phe}$ is constructed.

In $\mathbf{H}_{top}$, each node is connected with its topological neighbors in the image space, i.e., nearest patch belongs to the

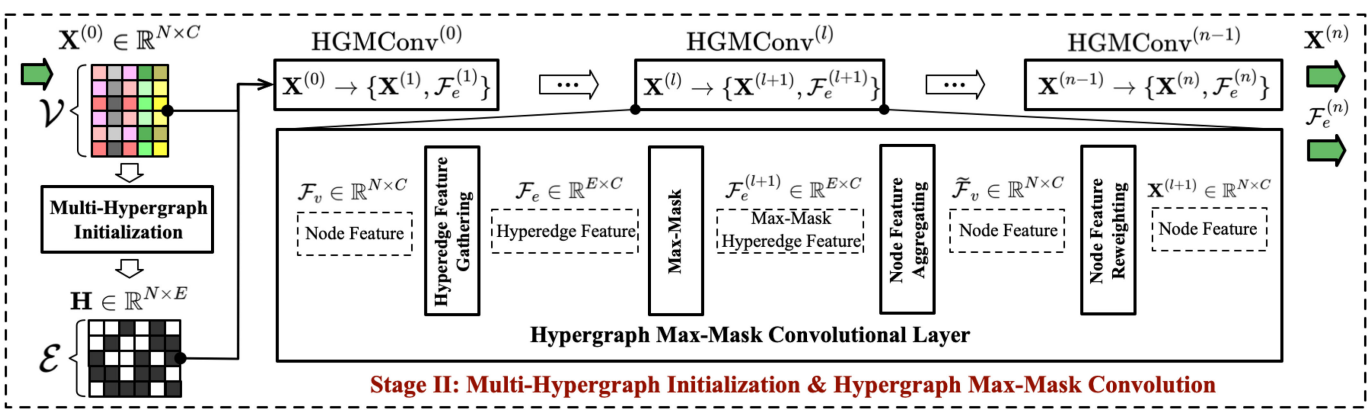

Fig. 5. Architecture of the multi-hypergraph learning stage; structure of a specific hypergraph max-mask convolutional layer HGMConv$^{(l)}$ is illustrated on the bottom.

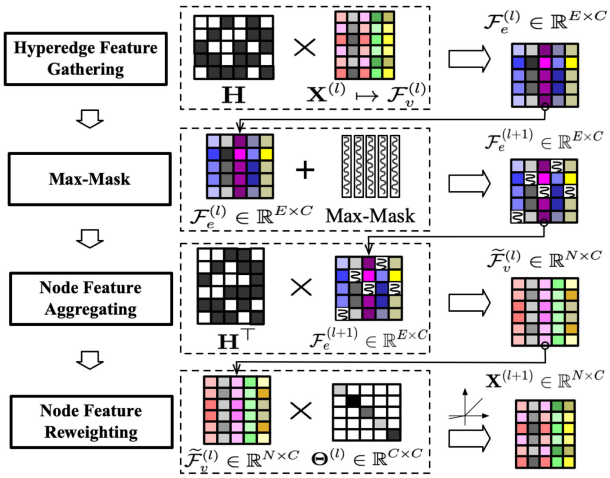

Fig. 6. Structure of a hypergraph max-mask convolutional layer.

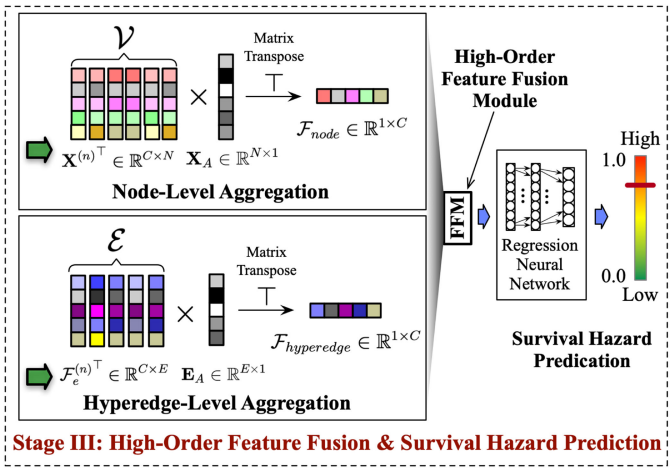

Fig. 7. Illustration of the third stage: Node- and Hyperedge- level feature aggregation.

same topological path. In our implementation, the initializing strategy of phenotype-wise sub-hypergraph includes the boundaries ROIs, $\mathbb{B}^{\frac{1}{4}}$, $\mathbb{B}^{\frac{1}{2}}$, $\mathbb{B}^{\frac{3}{4}}$, $\mathbb{B}^{1}$, and the centers of ROIs. The $E := |\mathcal{E}|$ hyperedges among $N := |\mathcal{V}|$ nodes are indicated by an incidence matrix $\mathbf{H} \in \mathbb{R}^{N\times E}$, representing a binary $0/1$ node-edge relation. The element in this incidence matrix is defined as

$$h(v,e) = \begin{cases} 1, & v \in e \\ 0, & v \notin e \end{cases} \tag{2}$$

where v and e denote node and hyperedge, respectively. Each hyperedge $\mathcal{E}$ provides a communication channel for the relevant patches and may corresponds to a pathogenic pattern.

After combining two sub-hypergraph incidence matrices to generate the incidence matrix $\mathbf{H}$ of the multi-hypergraph, the multi-hypergraph Laplacian $\mathbf{L}$, i.e., the normalized positive semi-definite Laplacian matrix of the resulting hypergraph, can be obtained by

$$\mathbf{L} = \mathbf{I} - \mathbf{D}_v^{-\frac{1}{2}}\mathbf{H}\mathbf{W}\mathbf{D}_e^{-1}\mathbf{H}^{\top}\mathbf{D}_v^{-\frac{1}{2}}, \tag{3}$$

where $\mathbf{I} \in \mathbb{R}^{N\times N}$ is an identity matrix. $\mathbf{D}_e \in \mathbb{R}^{E\times E}$, $\mathbf{D}_v \in \mathbb{R}^{N\times N}$ and $\mathbf{W} \in \mathbb{R}^{1\times E}$ denote the diagonal degree matrix of hyperedges, the degree matrix of nodes and weight matrix of hyperedges, respectively, obtained by

$$\begin{cases} \mathbf{D}_e = [\delta(e_i)], i \in [1,E] \\ \delta(e_i) = \sum_{v_j\in\mathcal{V}} h(v_j,e_i) \\ \mathbf{D}_v = [d(v_j)], j \in [1,N] \\ d(v_j) = \sum_{v_j\in e_i, e_i\in\mathcal{E}} w(e_i)h(v_j,e_i) \\ \mathbf{W} = [w(e_i)] \end{cases} \tag{4}$$

3.2.2 Hypergraph Max-Mask Convolution

The structure of the hypergraph max-mask convolution is shown in Fig. 7. It uses the spectral hypergraph convolutional layers $\mathrm{HGconv}(\cdot)$ in [20] as the backbone. The convolutional operation in each layer includes four steps as shown in Fig. 6. First, the node features $\mathcal{F}_v^{(l)}$ from the input signal $\mathbf{X}^{(l)}$ connected by each hyperedge are integrated to form hyperedge-level feature vectors $\mathcal{F}_e^{(l)} \in \mathbb{R}^{E\times C}$. This step is called "hyperedge feature gathering," in Fig. 7, implemented by the multiplication of $\mathbf{H}$ and $\mathcal{F}_v^{(l)}$ (see the top row of Fig. 6). After that, we perform a max-mask operation on each dimensionality of $\mathcal{F}_e^{(l)}$. The max-mask operation does not take the contribution of λ dominant hyperedges that take the largest values into account. Since each column of both $\mathcal{F}_v^{(l)}$ and $\mathcal{F}_e^{(l)}$ is an attribute vector corresponding to a specific latent factor affecting the survival status, this operation essentially avoids the situation that latent factors are learned from only a small number of dominant nodes (hyperedges), and thus alleviates the over-fitting issue. This step is called "max-mask" in Fig. 7 and is illustrated in the second row of Fig. 6

Next, in the step called "node feature aggregating," the output node features $\widetilde{\mathcal{F}}_v^{(l)}$ are obtained by aggregating the associated hyperedge features, which is achieved by multiplying matrix $\mathbf{H}^{\mathrm{T}}$ as shown in the third row of Fig. 6.

Lastly, the output node features $\widetilde{\mathcal{F}}_v^{(l)}$ are further weighted in a step called "node feature reweighting," by learnable parameters $\mathbf{\Theta}^{(l)}$ (a diagonal matrix), followed by a non-linear activation function $\sigma(\cdot)$ as shown in the bottom row of Fig. 6. Therefore, the formulation of a hypergraph max-mask convolutional layer $\mathrm{HGMconv}(\cdot)$ can be defined as

$$\begin{cases} \mathbf{X}^{(l+1)} = \sigma\Big[\Big((\mathbf{I}-\mathbf{L})\mathbf{X}^{(l)} + \mathbf{H}^{-1}(\mathbf{I}-\mathbf{L})\mathbf{X}^{(\lambda)}\Big)\mathbf{\Theta}^{(l)}\Big] \\ \mathcal{F}_e^{(l+1)} = \mathbf{H}^{-1}(\mathbf{I}-\mathbf{L})\mathbf{X}^{(l)} + \mathbf{X}^{(\lambda)} \end{cases} \tag{5}$$

where $\mathbf{X}^{(\lambda)}$ is an offset matrix only keeping the sparse negative values of the top λ dimensionalities for each attribute feature vector and leaving the others zero. $\mathbf{H}^{-1}(\mathbf{I}-\mathbf{L})\mathbf{X}^{(\lambda)}$ functionally makes top λ attribute feature dimensionalities in current layer ignored when calculating a gradient or updating the node features. $\sigma(\cdot)$ denotes the nonlinear activation function (e.g., $\mathrm{LeakyReLU}(\cdot)$). $\mathbf{\Theta}^{(l)}$ denotes the learnable parameters in the lth layer, serving like a MLP layer.

3.3 High-Order Survival Prediction

3.3.1 High-Order Feature Fusion

The high-order feature fusion module is to aggregate two level high-order features, i.e., node-level features $\mathbf{X}^{(n)}$ and hyperedge-level features $\mathcal{F}_e^{(n)}$, output from the hypergraph max-mask convolution module, to form a global representation for each WSI. Our aggregation strategy in this module

is intuitive and learnable as illustrated in the left of Fig. 7. We first perform aggregation for features from each level in parallel to form two global feature vectors and then concatenate the two global feature vectors into a single one by a feature fusion module (FFM). Specifically, for node-level features, we learn a global latent (attribute) feature vector $\mathcal{F}_{node}$, where the relative weight for each latent factor is obtained by the aggregation of all nodes (i.e., performing the multiplication of the transpose of $\mathbf{X}^{(n)}$ and a learnable weighting vector $\mathbf{X}_A$. We use a similar aggregation strategy for hyperedge-level features $\mathcal{F}_{hyperedge}$, i.e., perform the multiplication of the transpose of $\mathcal{F}_e^{(n)}$ and another learnable weighting vector $\mathbf{E}_A$. The FFM can be implemented in several approaches including mean fusion, max fusion, stochastic fusion [61], Lp fusion [62], *etc*.

3.3.2 Survival Hazard Prediction

The subsequent survival hazard prediction module is a regression network that takes the global high-order representation as input and outputs the final survival hazard score. In our implementation, we use multilayer perceptron (MLP) [63], [64], to regress the hazard score from the global high-order feature. The adopted nonlinear regression of MLP consists of three fully-connected layers as well as the *sigmoid* activation function. It is worth mentioning that the regression network can also be replaced by other widely-used approaches (e.g., Cox [30], En-Cox [34], BoostCI [44], LASSO [31]). Our experimental results in the next section will show that the adopted MLP can achieve relatively better overall performance.

In the entire framework, the hypergraph convolution, high-order feature fusion, as well as the survival hazard prediction modules can be trained in an end-to-end manner. Three kinds of losses can be used for the training. The first one is the mean squared error loss (MSE) defined as

$$\mathcal{L}_{MSE} = \frac{1}{P}\sum_{0}^{P}(h - \hat{h})^2, \tag{6}$$

where h and $\hat{h}$ denote the predicted hazard score and the ground truth hazard score of a WSI, respectively. P denotes the WSI patch number in a batch of the training procedure. The second loss function is the negative Cox log partial likelihood loss function adopted in DeepGraphSurv [5], formulated as follows:

$$\mathcal{L}_{NLL} = \sum_{i=1}^{M}\delta_i\left(-s_i^p + \log\sum_{j\in\{j:s_j^g\leq s_i^g\}}\exp(s_j^p)\right), \tag{7}$$

where s_i^p and s_i^g denote the predicted result and ground truth, respectively. $\mathcal{M}$ is the number of comparable pairs, derived from the number of patients. The third loss is the Bayesian Concordance Readjust (BCR) loss [15] which uses the concordance rate of pair-wise samples and point-wise prediction error reconstruction as the supervision. It includes $\mathcal{L}_{MSE}$ as its component and is defined as

$$\mathcal{L}_{BCR} = \left(\sum -\log\left(\delta(\mathbf{W}\cdot(\mathbf{X}_i^\star - \mathbf{X}_j^\star)^\top)\right)\right) + \mathcal{L}_{MSE}, \tag{8}$$

where $\mathbf{W}\in\mathbb{R}^{1\times C}$ is a linear weight vector. $\mathbf{X}_i^\star\in\mathbb{R}^{1\times C}$ and $\mathbf{X}_j^\star\in\mathbb{R}^{1\times C}$ are the output features from the last layer of the feature fusion module (FFM) for i_{th} and j_{th} WSI patches, respectively. δ denotes the sigmoid activate function. In next experimental section, the performances of these two losses will be evaluated and compared.

TABLE 1
Dataset Statistics

Mode	Sub-Set			Whole-Set		
Datasets	LUSC	GBM	NLST	LUSC	GBM	NLST
Patient Number	463	365	263	504	617	452
WSI Number	535	491	425	1612	2053	1225
Shortest Survival Time (days)	1	3	145	1	3	145
Longest Survival Time (days)	2620	3881	2624	5287	3881	2789

4 Experiments

4.1 Datasets and Baselines

We mainly evaluate the proposed approach on three datasets, including two lung cancer datasets (i.e., LUSC [24], NLST [25]) and a brain carcinoma dataset (i.e., GBM [24]). Both LUSC and GBM are from the generic cancer patient dataset TCGA [24]. There are different numbers of patients in these datasets and each patient has at least one WSI. The survive times of those patients vary from one to another. Table 1 summarizes the important statistical data for each datasets.

Data split in our experiments is performed in two types of settings. The first setting, termed as "Sub-Set," follows the experimental settings of previous works [4], [5], [7], and uses the same amount of WSI data. The other setting termed as "Whole-Set," employs all of the WSI data for the experiments. The supervision data, i.e., the normalized hazard score $\hat{h}_i$ for a WSI of a patient with survival duration T_i is computed by

$$\hat{h}_i = \frac{T_{min}\cdot(T_{max} - T_i)}{T_i\cdot(T_{max} - T_{min})}, \tag{9}$$

where T_{max}, T_{min} respectively denote the longest and shortest patient survival duration in each dataset. Besides regressing a hazard score within $[0, 1]$, survival prediction model also usually gives a binary prediction result to suggest if the subject belongs to a high- or low-risk group (i.e., a patient with a hazard score greater than the median score will be predicted as a high-risk patient).

We compare the proposed method with the following several baselines:

1) *DeepConvSurv* [13] is the first proposed CNN-based survival prediction model based on WSI. It takes the sampled patches ($224px \times 224px$) from the ROIs of whole-slide histopathological images as input and and predicts the survival hazard by Cox [30]. The ROIs are generated by the OTSU [59] based on the pigment threshold.
2) *WSISA* [4] is another CNN-based prediction model, which includes three stages, i.e., randomly sampling patches without the margin areas that contains few cells, extracting features by ResNet-50 [60] and predicting the survival hazard scores by Lasso-Cox [31].

3) *GCN* [22] is a general graph convolutional network based model. Same with WSISA, it takes the randomly sampled patches as input and extracts the low-level visual features first. Then, it builds up the graph structure by the euclidean distance threshold. After three layers of graph convolutional layers, the generated representation is fed into the final regression layer (MLP) and derives the final survival hazard score.
4) *DeepGraphSurv* [5] employs spectral GCN to take the topological relationships into consideration and then drops several less important patches by attention mechanism. We follow the same settings as illustrated in DeepGraphSurv [5], and input data is the same as WSISA. Its regression model employs the Cox regression model [30].
5) *DeepMISL* [6] considers multiple slides from one patient and prediction on the both local and global representations. We follow the experimental settings reported in DeepMISL [6]. The input is also same as WSISA, and the hazard score is predicted by MLP.
6) *RankSurv* [15] is a metric-driven pair-wise ranking based network. It adopts traditional hypergraph neural network [20] to achieve a high-order representation for survival prediction on WSI data.
7) *Patch-GCN* [23] uses a context-aware, spatially-resolved patched based graph convolutional network to model both local and global level topological structures in the tumor micro-environment by hierarchically aggregating instance-level histology features. Its feature extractor uses pre-trained ResNet-50 and the cross entropy-based Cox proportional loss function [65] is adopted in the last hidden layer.

4.2 Evaluation Metrics

Three evaluation metrics are adopted to measure the prediction accuracy.

- C-index [66]. C-index is commonly used to measure the model's ability to correctly provide a reliable ranking of the survival times based on the individual risk scores. C-index can be computed by

$$\mathcal{C} = \frac{1}{\mathcal{M}} \sum_{i:\delta_i=1} \sum_{j:T_i<T_j} \mathbb{1}[(T_i, X_i) < (T_j, X_j)], \tag{10}$$

  where $\mathcal{M}$ is the number of comparable pairs. $\mathbb{1}[.]$ denotes the indicator function. T_i (T_j) denotes the actual observed individual risk score and X_i (X_j) is the predicted risk score. The value of C-index ranges from 0 to 1. Larger C-index values correspond to better prediction performance and vice versa. 0 is the worst condition, 1 is the best, and 0.5 is the value as a random prediction.
- Receiver Operating Characteristic (ROC) [67]. ROC curve is another standard tool for the purpose of assessing the diagnostic ability of a binary classifier as its discrimination threshold is varied. The ROC curve can be created by plotting the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings. An ROC space is defined by FPR and TPR as x and y axes, respectively, which depicts relative trade-offs between true positive and false positive. A random guess would give a point along a diagonal line from the left bottom to the top right corners. The diagonal divides the ROC space. Points above the diagonal represent good classification results (better than random), points below the line poor results (worse than random).
- KM-estimation [26] is a tool which reports the proportion of survival patients within continuous tracking time. It can also give an intuitive visual performance comparison of binary classifiers, i.e., a more distinctive gap between the two curves for the low- and high-risk groups indicates a more accurate classification. KM-estimation curve can be calculated with the following equation

$$\widehat{S}(t) = \prod_{i:t_i \le t} \left(1 - \frac{d_i}{n_i}\right), \tag{11}$$

  where t_i denotes the time when at least one event (death) happened, d_i is the number of deaths that happened at t_i, and n_i is the individuals known to have survived (have not yet had an event or been censored) up to time t_i.

4.3 Implementation

In the patch sampling step, we adopt a public toolbox called "openslide" released in [68], and extract the patches with the size of $224px \times 224px$ at $20\times$ objective magnification level. In total, for each WSI we collect 2,000 patches from the yielded ROIs, following the same the setting with previous work [4], [5], [7], [15]. The low-level visual feature of each patch is extracted by ResNet-50 pre-trained on ImageNet [8], whose dimension is $x_i \in \mathbb{R}^{1\times512}$. The hypergraph vertex matrix $\mathbf{X} \in \mathbb{R}^{2000\times512}$ is built up by vertically stacking x_i. We conduct 10-fold cross-validation to comprehensively evaluate the proposed method and the comparison methods. The original WSI samples from each dataset are randomly partitioned into 10 equal-sized groups. Of the 10 groups, a single group is retained as the validation data for testing the model, and the remaining 9 groups are used as training data. The cross-validation process is then repeated 10 times, with each of the 10 groups used exactly once as the validation data. The 10 results can then be averaged to produce the final estimation. To select the optimal parameter $\mathcal{K}$ for the K-nearest neighbors (KNN) algorithm [69] in the generation of phenotype-wise sub-hypergraph and to study the influence of different $\mathcal{K}$ values, we conduct the experiments with $\mathcal{K}$ increasing from 5 to 20. We conduct 10-fold cross-validation on the training set and find $\mathcal{K} = 9$ gives the best overall results. Fig. 9 shows the whole validation results on all the three datasets where we can see the model performance of the adopted method HGSurvNet is not sensitive with respect to $\mathcal{K}$.

The number of hypergraph max-mask convolutional layers is set as 3. The λ in Eq. (5) is set to $E \times 0.25$ (E is the dimension of $\mathcal{F}_e^{(l)}$). For training, we use stochastic gradient descent with momentum 0.9 and weight decay 5×10^{-4} with a mini-batch size of 32. The learning rate is set to 10^{-4} at first, and then decreased to 10^{-5} after 50 epochs. In the

TABLE 2
Prediction Accuracy Comparison on the Whole-Set of LUSC, GBM, and NLST

Methods (metric: C-index)		LUSC		GBM		NLST	
DeepConvSurv [13]	*(mean, std, p-value)*	0.5882	*±0.015, 9.1847e-5*	0.5078	*±0.013, 8.1752e-5*	0.5033	*±0.015, 1.3724e-5*
WSISA [4]	*(mean, std, p-value)*	0.5976	*±0.013, 2.8761e-4*	0.5689	*±0.019, 8.7682e-4*	0.5536	*±0.016, 5.7121e-5*
GCN [22]	*(mean, std, p-value)*	0.6123	*±0.017, 3.1414e-3*	0.6044	*±0.011, 1.4273e-3*	0.5910	*±0.010, 5.3876e-4*
DeepGraphSurv [5]	*(mean, std, p-value)*	0.6164	*±0.012, 1.3138e-3*	0.6177	*±0.015, 4.1625e-2*	0.6212	*±0.018, 1.7253e-3*
DeepMISL [6]	*(mean, std, p-value)*	0.6235	*±0.024, 4.2732e-3*	0.6165	*±0.013, 1.8734e-2*	0.6354	*±0.017, 1.8431e-2*
RankSurv [15]	*(mean, std, p-value)*	0.6608	*±0.011, 4.7813e-2*	0.6627	*±0.011, 3.1180e-2*	0.6820	*±0.011, 3.1791e-2*
Patch-GCN [23]	*(mean, std, p-value)*	0.6227	*±0.013, 2.4892e-2*	0.6203	*±0.025, 3.1867e-2*	0.6332	*±0.016, 4.2871e-2*
HGSurvNet (Ours)	*(mean, std, -)*	**0.6730**	*±0.012, -*	**0.6726**	*±0.009, -*	**0.6901**	*±0.010, -*

The comparison is measured by the metric of C-index (mean, std, and p-value). Note that all of the p-values are much less than 0.05, which indicates the C-index based comparisons are statistically significant. The accuracy of the baselines was predicted by our own implementation on the same randomly sampled patches.

comparison with the baseline methods, we use the architecture which employs stochastic fusion [61] for node- and hyperedge- level feature fusion and MLP for the regression network since these components provide the best performances. For fair comparison, we compare HGSurvNet with other methods under the same setting of the loss function. In the pre-processing stage, ResNet-50 was used to extract the visual features of 2,000 patches for our method and its comparison. We ran the training for the proposal model with 8 GeForce RTX 1080 Ti GPUs for 1000 epochs. The training converges after around 8 hours (700 epochs), and the inference of each WSI took less than 35 milliseconds. In contrast, the GNN-based methods [5], [22] took about 14 hours, 6 hours more than our method, for the same settings. The CNN-based methods [4], [6], [13] were about 4 hours faster than our method but performed worse as shown in Table 2.

4.4 Results and Analysis

Performance comparison measured by C-index on the settings of Whole-Set and Sub-Set are respectively summarized in Tables 2 and 3. The statistical comparison on 10 repeated trials is shown in Fig. 8. It is worth mentioning that all of the baseline methods have not released their source codes, we use the results reported in those original studies for the comparison on the setting of Sub-Set. On the setting of Whole-Set, we report the results based on our own implementation for those baseline methods. Based on the quantitative results measured by C-index, we can see CNN-based models (i.e., DeepConvSurv, DeepCorrSurv, WSISA) generally have inferior performance than those graph structure-based methods (i.e., GCN, DeepGraphSurv, Patch-GCN, HGSurvNet). For example, on the setting of Whole-Set of NLST, GCN, DeepGraphSurv, Patch-GCN, RankSurv, and HGSurvNet achieve gains of 3.74%, 6.76%, 7.93%, 12.84%, and 13.65% compared with WSISA, respectively. The result shows that the high-order information captured by the graph structure can improve the performance of survival prediction. These quantitative results also clearly demonstrate that HGSurvNet outperforms the other competitors across all three datasets. For example, compared with GCN, HGSurvNet achieves gains of 7.92%, 7.27%, and 8.79% on the Sub-Set settings of LUSC, GBM, and NLST. Regarding the Whole-Set of LUSC, GBM, and NSL, HGSurvNet achieves gains of 6.07%, 6.82%, and 9.91%.

TABLE 3
Prediction Accuracy Comparison on the Metric of C-Index on the Sub-Set of LUSC, GBM, and NLST

Methods (metric: C-index)	LUSC	GBM	NLST
DeepConvSurv [13]	0.5715	0.5184	0.5367
WSISA [4]	0.6361	0.5609	0.6284
GCN [22]	0.6239	0.6153	0.6415
DeepGraphSurv [5]	0.6389	0.6344	0.6582
DeepMISL [6]	0.6518	0.6567	0.6694
RankSurv [15]	0.6791	0.6722	0.7183
Patch-GCN [23]	0.6435	0.6379	0.6571
HGSurvNet (Ours)	**0.7031**	**0.6880**	**0.7294**

Note that the values of std and p-value of C-index is not reported because of the lack of the experimental data of the baseline methods. The accuracy of the baselines was predicted by our own implementation.

Our method HGSurvNet achieves gains of 5.96%, 5.01%, and 7.23% on the Whole-Set of LUSC, GBM, and NLST compared with the most recent work Patch-GCN [23], respectively. The results indicate that the features based on high-order pathological interactions among global tissues and high-level topological patterns, learned by our method, may have a more significant effect than context-aware features produced from instance-level local neighborhoods graphs in Patch-GCN. It is worth mentioning that the quantitative comparison with DeepConvSurv [13] may not be fair because the results for DeepConvSurv [13] were produced under a setting where patches were sampled from those pathologist-annotated regions rather than randomly sampled from the WSIs excluding the non-tissue regions in our and other methods.

Fig. 10 compares HGSurvNet with two other state-of-art methods by KM-estimation curves in terms of the capability of binary risk classification, i.e., separating the high- from low- risk groups, on both train and validation sets of the Whole-Set settings of LUSC, GBM and NLST. We can clearly see that, on all of the validation sets, HGSurvNet has the most significant gap between the curves of the low- and high- risk groups, although both DeepGraphSurv and DeepMISL have close gaps as HGSurvNet on all of the train sets. It indicates that HGSurvNet can classify low- and high- risk patients more effectively. Moreover, we can also see that HGSurvNet shows closer performances on the train and validation sets than DeepGraphSurv and DeepMISL. This may be because the hypergraph max-mask convolution based network design in HGSurvNet alleviates the overfitting problem. Next, we study how much each technical

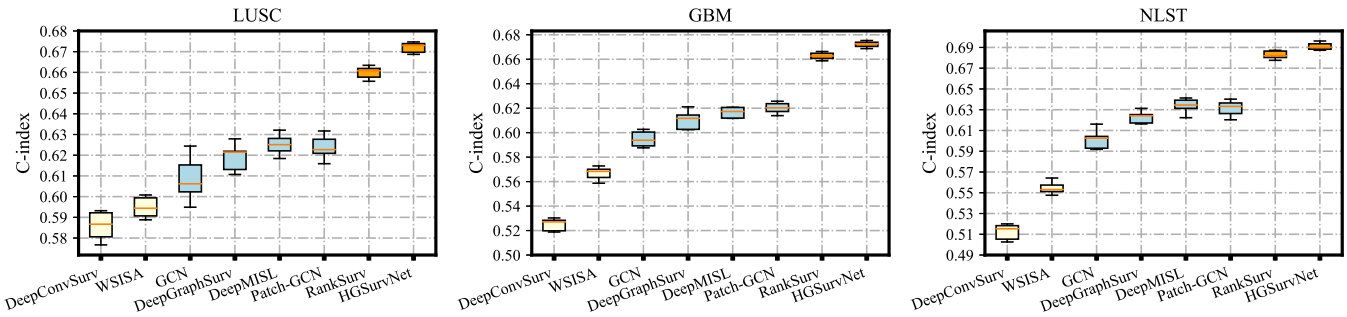

Fig. 8. The statistical comparison of different methods on the Whole-Set of LUSC, GBM, and NLST; the comparison is measured by C-index.

component in HGSurvNet contributes to the prediction performance improvement.

4.5 Ablation Study and Analysis

In this section, we conduct experiments on the Whole-Set of LUSC, GBM, and NLST to investigate the individual contributions of different components in the proposed HGSurvNet.

4.5.1 Study on Aggregation Strategy

This set of experiments investigate how the model performs under different aggregation strategies: HGSurvNet-node (only with node-level aggregation), HGSurvNet-hyperedge (only with hyperedge-level aggregation), and the adopted one fusing both node- and hyperedge- level representations. The results is shown in Table 4. It shows that HGSurvNet-node performs better than HGSurvNet-hyperedge on the two lung cancer datasets, NLST and LUSC, however, HGSurvNet-hyperedge demonstrates a stronger capability in processing the brain cancer data in GBM than HGSurvNet-node, i.e., HGSurvNet-hyperedge and HGSurvNet-node show their own advantages on the datasets of different cancers. Moreover, we can also see that the adopted aggregation strategy consistently outperforms either HGSurvNet-node or HGSurvNet-hyperedge on all of the three datasets by comparing the results using the same regression model. It indicates the proposed fusion strategy can exploit the complementary information in the node- and hyperedge- level representations and provide more accurate prediction.

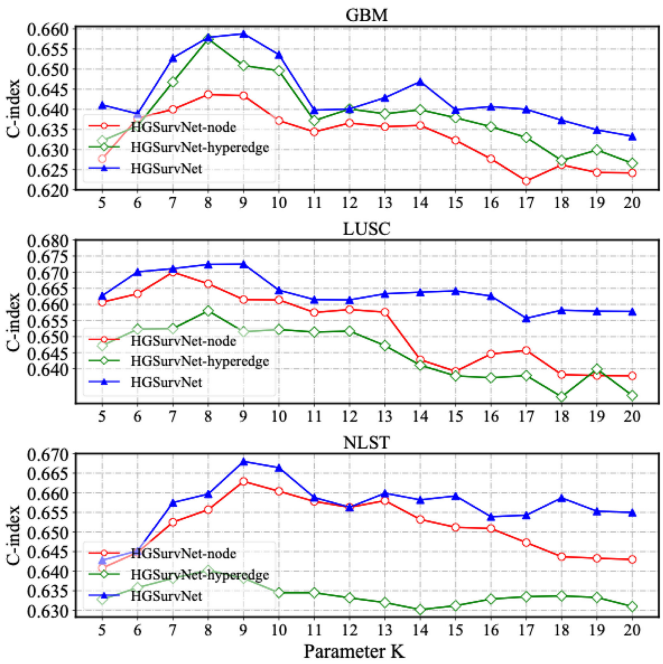

Fig. 9. Model performance under differing selected values for parameter K (for a patient having multiple WSIs, the predicted hazard score is the average of the scores on all those WSIs).

4.5.2 Study on Hypergraph Max-Mask Convolution

This set of experiments investigate how different settings of the parameter λ affects the representation. We conduct the experiments on all of the datasets from $\lambda = 0$, where the hypergraph max-mask convolutional layer degenerates into a typical hypergraph convolutional layer without any masked information. And then we gradually increase λ by $0.05 \times E$, and find the model performance goes up consistently on all of the datasets and peaks around $0.25 \times E$, and then it keeps dropping until the experiments end up at $0.40 \times E$. The experimental result measured by C-index is summarized in Fig. 11. It indicates that masking a certain number (i.e., top 25%) of dominant attribute feature dimensionalities can generally improve the prediction accuracy. However, a too small signal-to-mask ratio (i.e., large λ) would lead to the decline of prediction performance. Comparing with hypergraph convolutional convolution (i.e., $\lambda = 0$), we can see the proposed hypergraph max-mask convolution can improve the C-index by around 2% at most. From the ROC curves visualized in Fig. 12, we can see that the ROC curves are nearly unchanged when the signal-to-mask ratio decreases on the train sets of all three datasets (the red curves). However, the ROC curves change much more significantly on the validation sets. The ROC curves demonstrate the best prediction performances around $\lambda = 0.25 \times E$. This confirms our observation for the C-index based results.

4.5.3 Study on Multi-Hypergraph Learning

This set of experiments study if multi-hypergraph based learning can gain higher prediction accuracy than single hypergraph based learning. Besides the adopted multi-hypergraphs shown in Fig. 5, we construct two other types of architectures. In the first architecture, referred to as $\mathbf{H}_{phe}$, we remove the topology-wise sub-hypergraphs and only keep the phenotype-wise sub-hypergraph. In the other one, referred to as $\mathbf{H}_{top}$, we remove the phenotype-wise sub-hypergraph and only keep the topology-wise sub-hypergraphs. These results of these two types of architectures, together with the adopted multi-hypergraph based architecture referred to as $\mathbf{H}$, on all three datasets are summarized in Table 5. We can

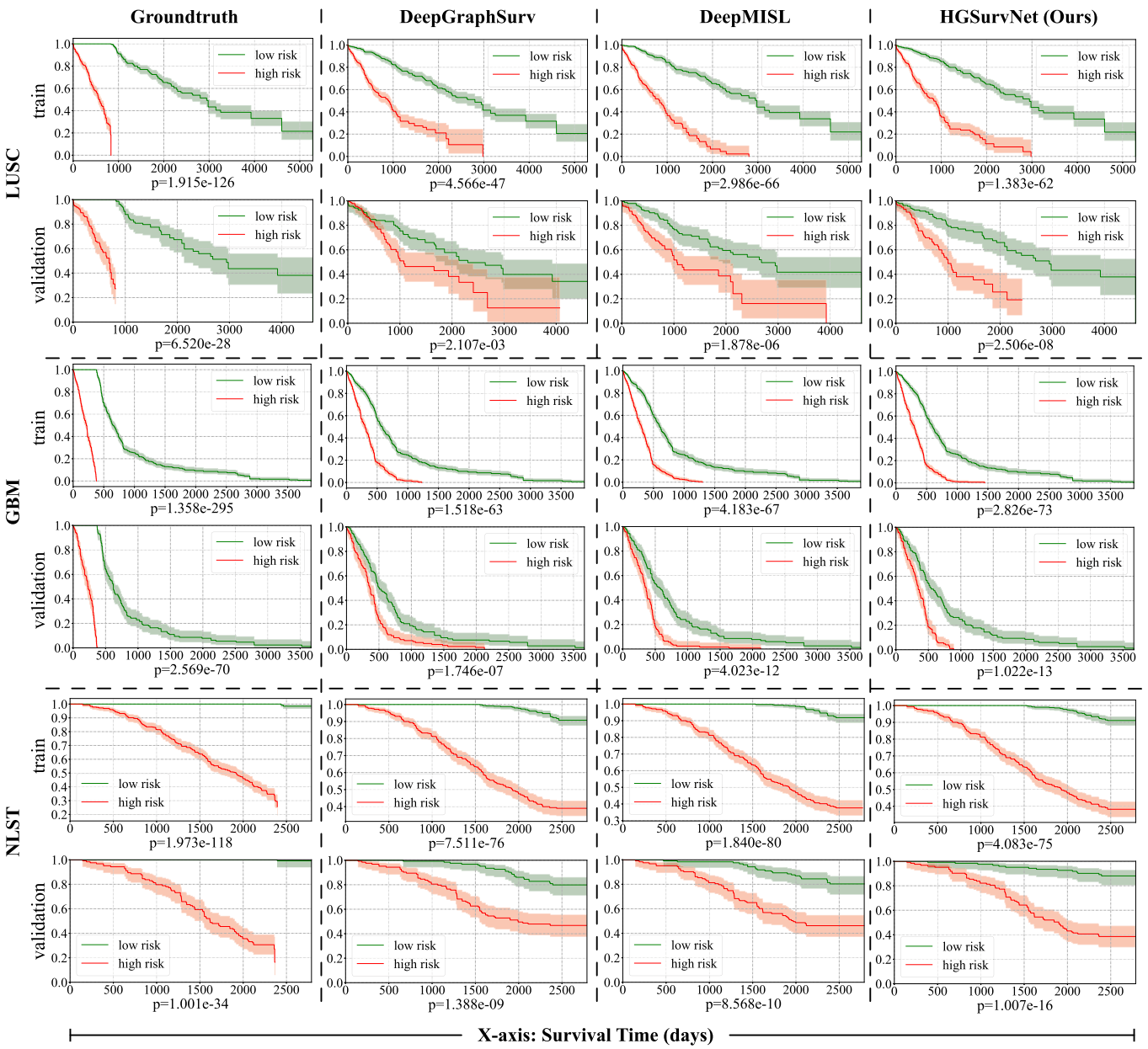

Fig. 10. HGSurvNet versus DeepGraphSurv versus DeepMISL, measured by KM-estimation curves, on train and validation sets of the Whole-Set of LUSC, GBM, and NLST. In the ground truth figures, low risk curves start going down from the probability of 1 at the median of the statistical survival time when the high-risk patients have been censored; more distinguish gaps between high- and low- risk curves correspond to better classification performance.

see $\mathbf{H}_{phe}$ outperforms $\mathbf{H}_{top}$ by 2% to 3% on all of the three datasets. This may be because $\mathbf{H}_{phe}$ takes use of more patch samples than $\mathbf{H}_{top}$ and thus is more representative. $\mathbf{H}$ consistently outperforms either $\mathbf{H}_{phe}$ or $\mathbf{H}_{top}$ on all of the three datasets. It indicates that the spatial structural information is complementary to the visual appearance information. More importantly, it implies the potential of the proposed multi-hypergraph learning framework in integrating complementary information from different sub-hypergraphs.

4.5.4 Study on Regression Model and Feature Fusion

This set of experiments studies which regression model and feature fusion are more powerful for the proposed framework. We mainly investigate and compare four classical regression models including three semi-parametric regularized Cox models commonly used in survival prediction, i.e., LASSO-Cox [31] and boosting cox model (Cox-boost) [44] and the elastic-net penalized Cox model (En-Cox) [34] as well as multi-layer perceptron (MLP). For feature fusion module, i.e., FFM in Fig. 7, we investigate four different feature fusion modules including Mean fusion, Max fusion, Stochastic fusion [61], and LP fusion [62]. Therefore, there are 4×4 different types of combination of regression and feature fusion modules in total. Table 4 shows the results of different regression modules under Stochastic fusion on all of the three datasets. We can see that MLP outperforms all of the other regression models by a significant margin. Specifically, compared with LASSO-Cox, Cox-boost, and En-Cox, MLP in the proposed HGSurvNet respectively improves the C-index-measured regression accuracy by about 4%, 3% and 2% on all of the three datasets. This may be

TABLE 4
Performance Comparison, Measured by C-Index, of Different Regression Models and Aggregation Strategies on the Whole-Set ("†" Denotes Significance Level is Reached as $p - Value < 0.05$)

Models	Regression	LUSC	GBM	NLST
HGSurvNet-node	**LASSO-Cox**	0.6278	0.5925	0.6224
	Cox-boost	0.6266	0.6097	0.6279
	En-Cox	0.6315	0.6196	0.6282
	MLP	**0.6700**†	**0.6437**†	**0.6629**†
HGSurvNet-hyperedge	**LASSO-Cox**	0.6033	0.6177	0.6033
	Cox-boost	0.6119	0.6259	0.6178
	En-Cox	0.6128	0.6273	0.6170
	MLP	**0.6580**†	**0.6579**†	**0.6402**†
HGSurvNet	**LASSO-Cox**	0.6291	0.6187	0.6291
	Cox-boost	0.6344	0.6294	0.6307
	En-Cox	0.6391	0.6301	0.6372
	MLP†	**0.6730**	**0.6726**	**0.6901**

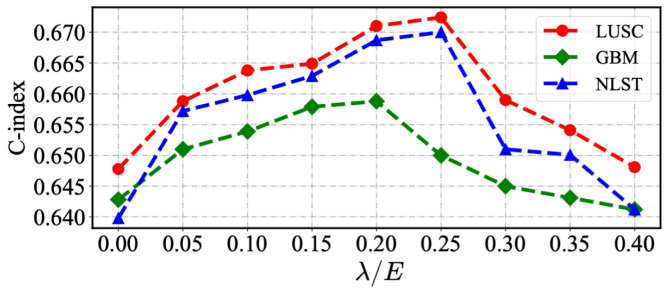

Fig. 11. Predication performances under different signal-to-mask ratios on LUSC, GBM, and NLST; λ is the number of the masked dominant dimensionalities and E is the feature length of $\mathcal{F}_e^{(l)}$.

because three fully-connected layers based multi-layer perceptron is more powerful than the Cox models where the size of model parameters is close to a one-layer MLP regressor. Table 6 suggests that the four different feature fusion modules do not demonstrate significant differences on the three datasets. In general, Stochastic fusion shows a slight advantage, less than 1% on C-index, on all three datasets. These results indicate the regression module has a greater influence on regression accuracy than the feature fusion module in the entire framework.

4.5.5 Study on Feature Extractor and Loss Function

This set of experiments study how much backbone feature extractors and loss functions affect the overall regression accuracy.

We conduct the ablation experiments to study and compare VGG [12], the adopted ResNet [60], and two typical transformer backbones: Swin-Transformer [71] and ViT [70]. Results shown in Table 7 suggest that different feature extractors employed on WSI patches do not make significant differences to the global representation of WSI data, which demonstrates the generality and robustness of the proposed framework. We can also observe that the transformer models [70], [71] cannot outperform the adopted ResNet, although they have shown superiority over CNNs on various computer vision tasks, which may be caused by the significant domain gap between photographic images (e.g., those in ImageNet [8]) and WSIs. Another reason may be due to the propensity of transformer-based approaches to overfit on smaller datasets compared to traditional CNNs.

Recent research [15] finds out BCR loss is more powerful in the problem of survival prediction than MSE loss. Another

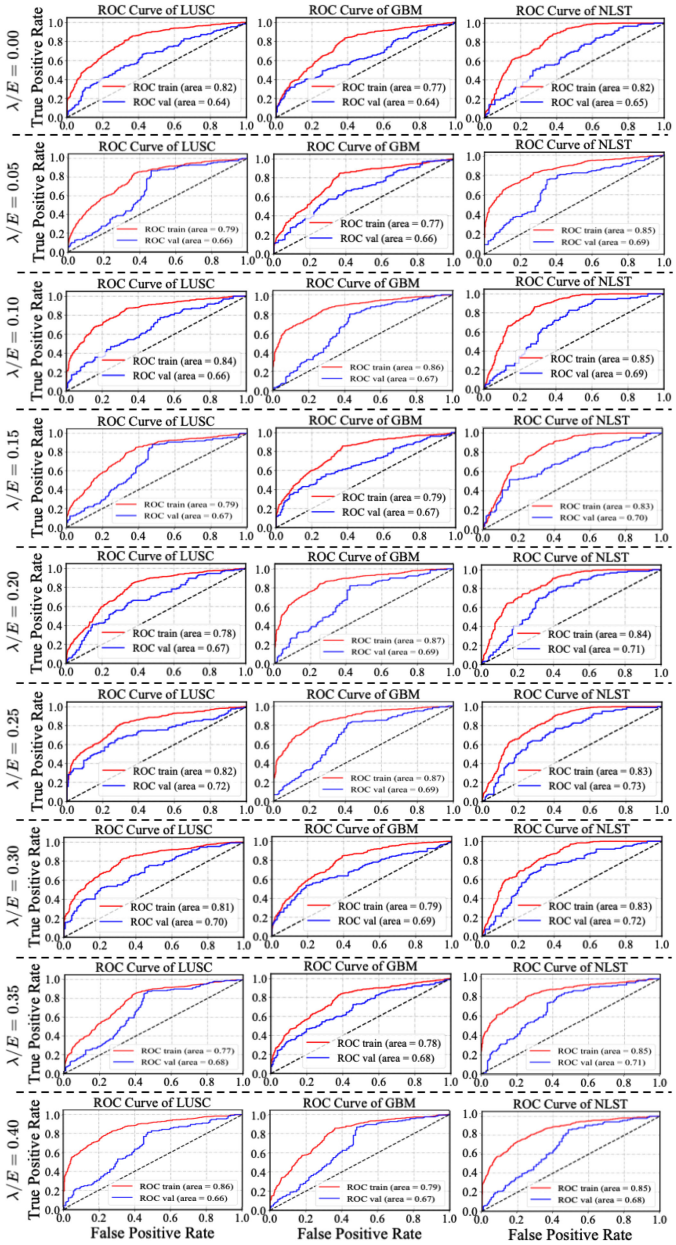

Fig. 12. From top to down: ROC curves of HGSurvNet under decreasing signal-to-mask ratios on LUSC, GBM, and NLST.

TABLE 5
Multi-Hypergraph versus Single-Hypergraph

Construction	LUSC		GBM		NLST	
$\mathbf{H}_{top}$	0.6331	± 0.041	0.6217	± 0.031	0.6477	± 0.028
$\mathbf{H}_{phe}$	0.6700	± 0.011	0.6502	± 0.026	0.6600	± 0.017
$\mathbf{H}$	**0.6730**	± 0.009	**0.6726**	± 0.007	**0.6901**	± 0.009

⋆ $\mathbf{H}_{top}$ *denotes the "topology-wise sub-hypergraph".*
⋆ $\mathbf{H}_{phe}$ *denotes the "phenotype-wise sub-hypergraph".*

TABLE 6
Performance Comparison of Different Feature Fusion Modules on the Whole-Set of LUSC, GBM, and NLST

FFM	LUSC		GBM		NLST	
Mean	0.6687	±0.007	0.6522	±0.009	0.6659	±0.013
Max	0.6724	±0.084	0.6586	±0.034	0.6677	±0.053
LP [62]	0.6720	±0.013	0.6580	±0.005	0.6667	±0.007
Stochastic [61]	**0.6730**	±0.009	**0.6726**	±0.007	**0.6901**	±0.009

TABLE 8
Performance Comparison of Loss Functions; Results are Measured by C-Index (Mean, Std)

Loss Function	LUSC		GBM		NLST	
NLL Loss	0.6713	±0.074	0.6573	±0.067	0.6677	±0.124
MSE Loss	0.6724	±0.012	0.6588	±0.019	0.6680	±0.014
BCR Loss	**0.6730**	±0.009	**0.6726**	±0.007	**0.6901**	±0.009

set of experiments is conducted to compare the performance differences between BCR loss and MSE loss within the proposed framework. We replace the MSE loss with BCR loss and re-train the network with the same experimental setting. The results shown in Table 8 suggest that the additional pair-wise concordance relation provided by the supervision data can further boost the prediction accuracy of the proposed framework.

4.5.6 Study on Interpretability

This set of experiments assessed the interpretability of our predictive model, i.e., determined if our model was helpful in assisting pathology diagnosis and reporting. Specifically, we determined if our model was able to locate pathological regions from the extremely large WSI slide, and identify their local cellular micro-environment as well as global distribution pattern. Our solution to achieve this is to extract the patches which contribute more information to the predicted hazard score, i.e., representative pathological patches, from the sampled patches of a WSI slide from the testing data-sets, and analyze if their micro-environment and global distribution pattern can support the diagnosis from the pathologist.

To locate these representative pathological patches, we turn to the latent attribute space of the WSI slides which are selected for the study. Specifically, we identify $M = 40$ patches with the greatest element values in each of the top $K = 5$ latent attribute vectors that are assigned the $K = 5$ highest weights in the last layer of the trained model. We extract and high-light $M \times K = 200$ representative pathological patches with different colors (i.e., red, orange, blue, cyan, and green, in the order of the significance of each latent attribute) for each slide as shown in Fig. 13. A pathologist was invited to make diagnoses for the selected WSI slides in this study.

From Fig. 13, we can see that the highlighted patches which contribute the most on the identical latent attribute share similar visual characteristic, e.g., nuclei are crowded to a similar degree, nucleoli are closely conspicuous, or the patches show similar level of pleomorphism indicating that our model is capable of summarizing the phenotype for micro-environment analysis similar to the pathologist's diagnosis. We also observed that the highlighted patches can also present the distribution pattern of the phenotype identified by each latent attribute. For example, in the LUSC slide with a survival time of 145 days from NLST, we can see red patches with a high grade of pleomorphism scatter in the patient's lung, By contrast, in the slide with a 570 day survival time, those red patches with relatively lower grade of pleomorphism seemingly cluster with severe degree of crowding in a local region inside the lung. Both of the cases correspond to higher hazard scores compared to the slide with a survival time of 2,789 days where representative patches scattering in the patient's lung have much lower grades of pleomorphism.

The visualized pathology information indicates that our model is capable of pathology region location, pathology region distribution pattern and intensity visualization, phenotype identification, and cell micro-environment analysis, which is crucial for pathology diagnosis and reporting.

TABLE 7
Performance Comparison of Different Visual Feature Extractors; Results are Measured by C-Index (Mean, Std)

Extractor	LUSC		GBM		NLST	
VGG [12]	0.6707	±0.011	0.6670	±0.010	0.6859	±0.009
ResNet [60]	**0.6730**	±0.009	**0.6726**	±0.007	**0.6901**	±0.009
ViT [70]	0.6703	±0.032	0.6679	±0.037	0.6828	±0.019
Swin-Transformer [71]	0.6720	±0.028	0.6691	±0.016	0.6882	±0.015

5 Discussion, Conclusion and Future Work

This paper proposes a multi-hypergraph end-to-end learning framework called HGSurvNet for survival prediction. HGSurvNet takes the whole slide images of a patient as input and regresses a hazard score indicating his/her life duration. HGSurvNet demonstrates consistent superior performances and outperforms the state-of-the-art methods by a large margin on two lung cancer datasets, LUSC and NLST, and a brain carcinoma dataset GBM. Coupled with the Bayesian Concordance Readjust (BCR) loss using the concordance rates of pair-wise samples as supervision, HGSurvNet can effectively improve the prediction accuracy on LUSC, GBM, and NLST, respectively. This is achieved by generating an effective high-order global feature representation, which is difficult to achieve for previous point-wise or pair-wise relation based learning frameworks, through two major technical contributions: a multi-hypergraph based high-order representation learning framework and a general hypergraph convolution network. The latter has the ability to alleviate over-fitting issue caused by a limited amount of training data. The network is significance when applying hypergraph based learning frameworks to WSI data because medical privacy is a major concern for many patients.

We wish to extend this work in several ways. First, we have conducted experiments to demonstrate that low-level feature extractors pre-trained on natural image datasets (ImageNet), are capable of extracting visual features for WSIs. Related works [4], [6], have verified that these pre-trained VGG or ResNet models are capable of feature extraction involving complex tissue patterns. Noting the successes of pre-trained models on large-scale datasets, such as BERT [72], XLNet [73], VideoBERT [74] *etc.*, we believe the pre-training on a large-

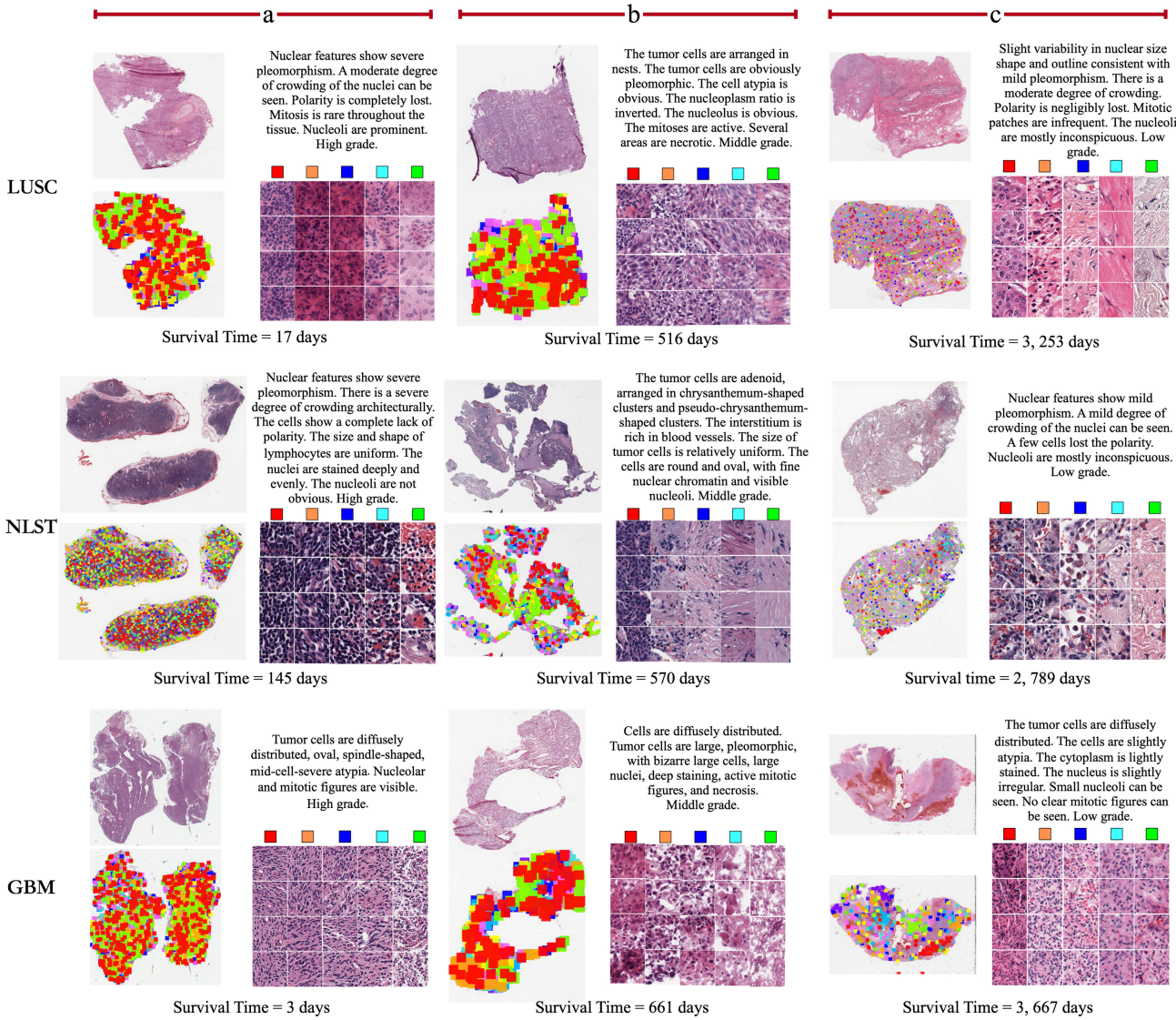

Fig. 13. Visualization of interpretable predictions of the method. a-c show the pathologist's descriptions, representative pathological patches, i.e., phenotype, located by latent attribute analysis, and the pathology patch distribution of three representative WSI.

scale histopathology-specific dataset is necessary and would boost related downstream tasks [75], [76], [77], [78]. It may be a practical approach to achieve this goal by randomly masking patches to encode and predict the corresponding pathology representation using contextual information. With a more powerful pathology-specific feature extractor, our framework may not only perform more effectively and stably on survival prediction but improve related downstream subtasks.

The current framework focuses on survival prediction using a single whole-slide image. This application scenario is common when the patients perform their first whole slide imaging. Study on this problem is meaningful because first-hand early diagnosis is critical for the treatment. Through insight analysis and comprehensive experiments, we see the effectiveness and general usefulness of the proposed framework, as well as the potential in applying it to a wider range of scenarios, including survival prediction, using multiple digitised whole-slide images (WSIs) of serial histology sections or even using multi-modality pathologic and clinical data.

ACKNOWLEDGMENTS

Donglin Di and Changqing Zou contribute equally to this work.

REFERENCES

[1] Z. Zhang et al., "Pathologist-level interpretable whole-slide cancer diagnosis with deep learning," *Nature Mach. Intell.*, vol. 1, no. 5, pp. 236–245, 2019.

[2] J. Gamper, N. A. Koohbanani, K. Benet, A. Khuram, and N. Rajpoot, "PanNuke: An open pan-cancer histology dataset for nuclei instance segmentation and classification," in *Proc. Eur. Congr. Digit. Pathol.*, 2019, pp. 11–19.

[3] D. J. Hartman, J. A. Van Der Laak, M. N. Gurcan, and L. Pantanowitz, "Value of public challenges for the development of pathology deep learning algorithms," *J. Pathol. Informat.*, vol. 11, 2020, Art. no. 7.

[4] X. Zhu, J. Yao, F. Zhu, and J. Huang, "WSISA: Making survival prediction from whole slide histopathological images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6855–6863.

[5] R. Li, J. Yao, X. Zhu, Y. Li, and J. Huang, "Graph cnn for survival analysis on whole slide pathological images," in *Proc. Med. Image Comput. Comput.-Assisted Interv.*, 2018, pp. 174–182.

[6] J. Yao, X. Zhu, and J. Huang, "Deep multi-instance learning for survival prediction from whole slide images," in *Proc. Med. Image Comput. Comput.-Assisted Intervention*, 2019, pp. 496–504.

[7] J. Yao, X. Zhu, F. Zhu, and J. Huang, "Deep correlational learning for survival prediction from multi-modality data," in *Proc. Med. Image Comput. Comput.-Assisted Interv.*, 2017, pp. 406–414.

[8] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 248–255.

[9] S. Lathuilière, P. Mesejo, X. Alameda-Pineda, and R. Horaud, "A comprehensive analysis of deep regression," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 9, pp. 2065–2081, Sep. 2019.

[10] Y. Kong and Y. Fu, "Human action recognition and prediction: A survey," 2018, *arXiv:1806.11230*.

[11] M. Adnan, S. Kalra, and H. R. Tizhoosh, "Representation learning of histopathology images using graph neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshop.*, 2020, pp. 4254–4261.

[12] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[13] X. Zhu, J. Yao, and J. Huang, "Deep convolutional neural network for survival analysis with pathological images," in *Proc. IEEE Conf. Bioinf. Biomed.*, 2016, pp. 544–547.

[14] D. Zhou, J. Huang, and B. Schölkopf, "Learning with hypergraphs: Clustering, classification, and embedding," in *Proc. Adv. Neural Inf. Process. Syst.*, 2006, pp. 1601–1608.

[15] D. Di, S. Li, J. Zhang, and Y. Gao, "Ranking-based survival prediction on histopathological whole-slide images," in *Proc. Med. Image Comput. Comput.-Assisted Interv.*, 2020, pp. 428–438.

[16] S. Zhang, S. Cui, and Z. Ding, "Hypergraph spectral clustering for point cloud segmentation," *IEEE Signal Process. Lett.*, vol. 27, pp. 1655–1659, 2020.

[17] F. Luo, B. Du, L. Zhang, L. Zhang, and D. Tao, "Feature learning using spatial-spectral hypergraph discriminant analysis for hyperspectral image," *IEEE Trans. Cybern.*, vol. 49, no. 7, pp. 2406–2419, Jul. 2019.

[18] Y. Zhang et al., "Hypergraph label propagation network," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 6885–6892.

[19] N. Franzese, A. Groce, T. M. Murali, and A. M. Ritz, "Hypergraph-based connectivity measures for signaling pathway topologies," *PLoS Comput. Biol.*, vol. 15, no. 10, 2019, Art. no. e1007384.

[20] Y. Feng, H. You, Z. Zhang, R. Ji, and Y. Gao, "Hypergraph neural networks," in *Proc. AAAI Conf. Artif. Intell.*, 2019, pp. 3558–3565.

[21] Y. Gao, Z. Zhang, H. Lin, X. Zhao, S. Du, and C. Zou, "Hypergraph learning: Methods and practices," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 5, pp. 2548–2566, May 2022.

[22] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2016, *arXiv:1609.02907*.

[23] R. J. Chen et al., "Whole slide images are 2 d point clouds: Context-aware survival prediction using patch-based graph convolutional networks," in *Proc. Med. Image Comput. Comput.-Assisted Interv.*, 2021, pp. 339–349.

[24] C. Kandoth et al., "Mutational landscape and significance across 12 major cancer types," *Nature*, vol. 502, no. 7471, pp. 333–339, 2013.

[25] B. S. Kramer, C. D. Berg, D. R. Aberle, and P. C. Prorok, "Lung cancer screening with low-dose helical ct: Results from the national lung screening trial (nlst)," *J. Med. Screening*, vol. 18, no. 3, pp. 109–111, 2011.

[26] E. L. Kaplan and P. Meier, "Nonparametric estimation from incomplete observations," *J. Amer. Statist. Assoc.*, vol. 53, no. 282, pp. 457–481, 1958.

[27] E. T. Lee and J. Wang, *Statistical Methods for Survival Data Analysis*, vol. 476, Hoboken, NJ, USA: Wiley, 2003.

[28] P. K. Andersen, O. Borgan, R. D. Gill, and N. Keiding, *Statistical Models Based on Counting Processes*, Berlin, Germany: Springer, 2012.

[29] S. J. Cutler and F. Ederer, "Maximum utilization of the life table method in analyzing survival," *J. Chronic Dis.*, vol. 8, no. 6, pp. 699–712, 1958.

[30] D. R. Cox, "Regression models and life-tables," *J. Roy. Statist. Soc.: Ser. B. (Methodol.)*, vol. 34, no. 2, pp. 187–202, 1972.

[31] R. Tibshirani, "The lasso method for variable selection in the cox model," *Statist. Med.*, vol. 16, no. 4, pp. 385–395, 1997.

[32] A. E. Hoerl and R. W. Kennard, "Ridge regression: Biased estimation for nonorthogonal problems," *Technometrics*, vol. 12, no. 1, pp. 55–67, 1970.

[33] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *J. Roy. Statist. Soc.: Ser. B. (Statist. Methodol.)*, vol. 67, no. 2, pp. 301–320, 2005.

[34] Y. Yang and H. Zou, "A cocktail algorithm for solving the elastic net penalized cox's regression in high dimensions," *Statist. Interface*, vol. 6, no. 2, pp. 167–173, 2013.

[35] Y. Li, K. S. Xu, and C. K. Reddy, "Regularized parametric regression for high-dimensional survival analysis," in *Proc. Soc. Ind. Appl. Math. Int. Conf. Data Mining*, 2016, pp. 765–773.

[36] L. Gordon and R. A. Olshen, "Tree-structured survival analysis.," *Cancer Treat. Rep.*, vol. 69, no. 10, pp. 1065–1069, 1985.

[37] P. J. Lisboa, H. Wong, P. Harris, and R. Swindell, "A bayesian neural network approach for modelling censored data with an application to prognosis after surgery for breast cancer," *Artif. Intell. Med.*, vol. 28, no. 1, pp. 1–25, 2003.

[38] M. J. Fard, P. Wang, S. Chawla, and C. K. Reddy, "A Bayesian perspective on early stage event prediction in longitudinal data," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 12, pp. 3126–3139, Dec. 2016.

[39] D. Faraggi and R. Simon, "A neural network model for survival data," *Statist. Med.*, vol. 14, no. 1, pp. 73–82, 1995.

[40] F. M. Khan and V. B. Zubek, "Support vector regression for censored data (SVRc): A novel tool for survival analysis," in *Proc. IEEE Int. Conf. Data Mining*, 2008, pp. 863–868.

[41] A. Widodo and B.-S. Yang, "Application of relevance vector machine and survival probability to machine degradation assessment," *Expert Syst. Appl.*, vol. 38, no. 3, pp. 2592–2599, 2011.

[42] F. Kiaee, H. Sheikhzadeh, and S. E. Mahabadi, "Relevance vector machine for survival analysis," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 3, pp. 648–660, Mar. 2015.

[43] H. Ishwaran, U. B. Kogalur, X. Chen, and A. J. Minn, "Random survival forests for high-dimensional data," *Statist. Anal. Data Mining: ASA Data Sci. J.*, vol. 4, no. 1, pp. 115–132, 2011.

[44] A. Mayr and M. Schmid, "Boosting the concordance index for survival data–a unified framework to derive and evaluate biomarker combinations," *PLoS One*, vol. 9, no. 1, 2014, Art. no. 84483.

[45] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," 2017, *arXiv:1710.10903*.

[46] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 1024–1034.

[47] P. Mobadersany et al., "Predicting cancer outcomes from histology and genomics using convolutional networks," *Proc. Nat. Acad. Sci.*, vol. 115, no. 13, pp. E2970–E2979, 2018.

[48] 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.

[49] R. J. Chen et al., "Pan-cancer integrative histology-genomic analysis via interpretable multimodal deep learning," 2021, *arXiv:2108.02278*.

[50] R. J. Chen et al., "Multimodal co-attention transformer for survival prediction in gigapixel whole slide images," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 3995–4005.

[51] C. Zu et al., "Identifying high order brain connectome biomarkers via learning on hypergraph," in *Proc. Int. Workshop Mach. Learn. Med. Imag.*, 2016, pp. 1–9.

[52] B. C. Munsell, G. Wu, Y. Gao, N. Desisto, and M. Styner, "Identifying relationships in functional and structural connectome data using a hypergraph learning method," in *Proc. Med. Image Comput. Comput.-Assisted Interv.*, 2016, pp. 9–17.

[53] D. Zhou, J. Huang, and B. Schölkopf, "Learning with hypergraphs: Clustering, classification, and embedding," in *Proc. Adv. Neural Inf. Process. Syst.*, 2007, pp. 1601–1608.

[54] M. Liu, J. Zhang, P.-T. Yap, and D. Shen, "View-aligned hypergraph learning for alzheimer's disease diagnosis with incomplete multi-modality data," *Med. Image Anal.*, vol. 36, pp. 123–134, 2017.

[55] P. Li and O. Milenkovic, "Inhomogeneous hypergraph clustering with applications," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 2308–2318.

[56] Y. Huang, Q. Liu, and D. Metaxas, "Video object segmentation by hypergraph cut," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 1738–1745.

[57] Y. Huang, Q. Liu, S. Zhang, and D. N. Metaxas, "Image retrieval via probabilistic hypergraph ranking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 3376–3383.

[58] L. Zhu, J. Shen, H. Jin, R. Zheng, and L. Xie, "Content-based visual landmark search via multimodal hypergraph learning," *IEEE Trans. Cybern.*, vol. 45, no. 12, pp. 2756–2769, Dec. 2015.

[59] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 9, no. 1, pp. 62–66, Jan. 1979.

[60] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[61] M. D. Zeiler and R. Fergus, "Stochastic pooling for regularization of deep convolutional neural networks," 2013, *arXiv:1301.3557*.
[62] A. Hyvärinen and U. Köster, "Complex cell pooling and the statistics of natural images," *Netw.: Comput. Neural Syst.*, vol. 18, no. 2, pp. 81–100, 2007.
[63] G. Thimm and E. Fiesler, "High-order and multilayer perceptron initialization," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 8, no. 2, pp. 349–359, Mar. 1997.
[64] U. Orhan, M. Hekim, and M. Ozer, "Eeg signals classification using the k-means clustering and a multilayer perceptron neural network model," *Expert Syst. Appl.*, vol. 38, no. 10, pp. 13475–13481, 2011.
[65] S. G. Zadeh and M. Schmid, "Bias in cross-entropy-based training of deep survival networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 9, pp. 3126–3137, Sep. 2021.
[66] P. J. Heagerty and Y. Zheng, "Survival model predictive accuracy and ROC curves," *Biometrics*, vol. 61, no. 1, pp. 92–105, 2005.
[67] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognit. Lett.*, vol. 27, no. 8, pp. 861–874, 2006.
[68] A. Goode, B. Gilbert, J. Harkes, D. Jukic, and M. Satyanarayanan, "Openslide: A vendor-neutral software foundation for digital pathology," *J. Pathol. Informat.*, vol. 4, no. 1, 2013, Art. no. 27.
[69] J. M. Keller, M. R. Gray, and J. A. Givens, "A fuzzy k-nearest neighbor algorithm," *IEEE Trans. Syst., Man, Cybern., Syst*, vol. SMC-15, no. 4, pp. 580–585, Jul./Aug. 1985.
[70] A. Dosovitskiy et al., "An image is worth 16x16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Learn. Representations.*, 2021.
[71] Z. Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 10012–10022.
[72] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*.
[73] Z. Yang et al., "XLNet: Generalized autoregressive pretraining for language understanding," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 5753–5763.
[74] C. Sun, A. Myers, C. Vondrick, K. Murphy, and C. Schmid, "VideoBERT: A joint model for video and language representation learning," in *Proc. Int. Conf. Comput. Vis.*, 2019, pp. 7464–7473.
[75] G. Litjens et al., "1399 h&e-stained sentinel lymph node sections of breast cancer patients: The camelyon dataset," *GigaScience*, vol. 7, no. 6, pp. 1–8, 2018.
[76] P. Bandi et al., "From detection of individual metastases to classification of lymph node status at the patient level: The camelyon17 challenge," *IEEE Trans. Med. Imag.*, vol. 38, no. 2, pp. 550–560, 2018.
[77] B. E. Bejnordi et al., "Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer," *J. Amer. Med. Assoc.*, vol. 318, no. 22, pp. 2199–2210, 2017.
[78] H.-Y. Zhou, S. Yu, C. Bian, Y. Hu, K. Ma, and Y. Zheng, "Comparing to learn: Surpassing imagenet pretraining on radiographs by comparing image representations," in *Proc. Med. Image Comput. Comput.-Assisted Interv.*, 2020, pp. 398–407.

Donglin Di received the BE degree from the Harbin Institute of Technology, Harbin, China. His research interests include medical image processing and machine learning.

Changqing Zou (Member, IEEE) received the BE degree from the Harbin Institute of Technology, Harbin, China, the ME degree from the Institute of Remote Sensing and Digital Earth, Chinese Academy of Sciences, Beijing, China, and the PhD degree from the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Beijing, China. His research interests include computer vision, computer graphics, and machine learning.

Yifan Feng received the BE degree in computer science from the Xidian University, Shaanxi, China, in 2018, and the ME degree in information science from Xiamen University, Fujian, China, in 2021. He is currently working toward the PhD degree with the School of Software, Tsinghua University, Beijing, China.

Haiyan Zhou received the PhD degree from the Central South University, China. She had presided over and participated in a number of national and provincial scientific research projects and teaching projects. She is the project director of standardized training for resident physicians with the Clinical Pathology Department of Xiangya Hospital, Central South University.

Rongrong Ji (Senior Member, IEEE) is currently a Nanqiang distinguished professor with Xiamen University, China, the deputy director of the Office of Science and Technology, Xiamen University, China, and the director of Media Analytics and Computing Lab. He was awarded as the National Science Foundation for Excellent Young Scholars (2014), the National Ten Thousand Plan for Young Top Talents (2017), and the National Science Fundation for Distinguished Young Scholars (2020). His research interests include computer vision, multimedia analysis, and machine learning. He has published more than 50 papers in *ACM/IEEE Transactions*, including the *IEEE Transactions on Pattern Analysis and Machine Intelligence* and the *International Journal of Computer Vision*, and more than 100 full papers on top-tier conferences, such as CVPR and NeurIPS. His publications have got more than 10 K citations in Google Scholar. He was the recipient of the Best Paper Award of ACM Multimedia 2011. He has served as area chairs in top-tier conferences such as CVPR and ACM Multimedia. He is also an advisory member for Artificial Intelligence Construction in the Electronic Information Education Commitee of the National Ministry of Education.

Qionghai Dai (Senior Member, IEEE) received the ME and PhD degrees in computer science and automation from Northeastern University, Shenyang, China, in 1994 and 1996, respectively. He has been the Faculty Member of Tsinghua University since 1997. He is currently a professor with the Department of Automation, Tsinghua University, Beijing, China, and the director of the Broadband and Digital Media Laboratory. His research areas include computational photography and microscopy, computer vision and graphics, and video communication. He is associate editor of the *Journal of Visual Communication and Image Representation*, the *IEEE Transactions on Neural Networks and Learning Systems*, and the *IEEE Transactions on Image Processing*. He is a member of the Chinese Academy of Engineering.

Yue Gao (Senior Member, IEEE) received the BE degree from the Harbin Institute of Technology, Harbin, China, and the ME and PhD degrees from Tsinghua University, Beijing, China. He is currently an associate professor with the School of Software, Tsinghua University.