Integration of Patch Features through Self-Supervised Learning and Transformer for Survival Analysis on Whole Slide Images

Ziwang Huang[1], Hua Chai[1], Ruoqi Wang[1], Haitao Wang[1], Yuedong Yang[1], and Hejun Wu[1]

School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, China
{wuhejun, yangyd25}@mail.sysu.edu.cn

Abstract. Survival prediction using whole slide images (WSIs) can provide guidance for better treatment of diseases and patient care. Previous methods usually extract and process only image features from patches of WSIs. However, they ignore the significant role of spatial information of patches and the correlation between the patches of WSIs. Furthermore, those methods extract the patch features through the model pre-trained on ImageNet, overlooking the huge gap between WSIs and natural images. Therefore, we propose a new method, called SeTranSurv, for survival prediction. SeTranSurv extracts patch features from WSIs through self-supervised learning and adaptively aggregates these features according to their spatial information and correlation between patches using the Transformer. Experiments on three large cancer datasets indicate the effectiveness of our model. More importantly, SeTranSurv has better interpretability in locating important patterns and features that contribute to accurate cancer survival prediction.

Keywords: WSI · Survival Analysis · Transformer · Self-Supervised Learning.

1 Introduction

Survival analysis generally refers to a statistical process that investigates the occurrence time of a certain event. Accurate survival analysis provides invaluable guidance for clinical treatment. For instance, the prognostic models in survival prediction can show the interactions between different prognostic factors in certain diseases. These results from survival prediction would allow clinicians to make early decisions on the treatment of diseases. Such early clinical interventions are crucial for the healthcare of patients.

There have been many computational methods proposed for survival analysis from whole slide images (WSIs) recently. Traditional methods generally

H. Wu and Y. Yang are co-corresponding authors. Z. Huang and H. Chai contributed equally to this work.

select several discriminative patches from manually annotated Region of Interests (RoIs) and then extract features for predictions [16–18, 20]. Relatively new methods [11, 14, 22] choose patches without using RoI annotations, as RoI annotations require heavy manpower. The previous methods that do not need RoI annotations usually adopt the ImageNet [13] pre-trained network to extract patch features from WSIs. For instance, WSISA proposed by Zhu et al. [22] extracts patches from the WSIs and gathers them into clusters of different patterns. WSISA adopts DeepConvSurv [21] to select meaningful patch clusters. These clusters are then aggregated for later prediction. Li et al. [11] propose a survival analysis method that first constructs a topological relationship between features and then learns useful representative and related features from their patches through a graph convolutional network [10] (GCN). However, extracting accurate patch features for survival analysis and integrating patch features to obtain an aggregated set of patient-level features constitute the two significant challenges. As stated, pre-training models overlook the huge gap between WSIs and natural images and no available labels can be used to fine-tune the pre-training models. As a result, the patch features from pre-trained models cannot satisfy the accuracy requirement for survival analysis. Additionally, previous methods do not notice the significant role of patch spatial information and the correlation between patches of WSIs. These methods usually separately process each cluster of patches or every single patch from the WSIs of the patient. This is due to the large scale of WSI, which makes it difficult to integrate spatial information into the model. Therefore, how to integrate patch features to obtain patient-level features is also an open question.

Recent studies have shown that SimCLR [1], a self-supervised learning method, can train a model with excellent feature extraction ability through contrastive learning. The feature extraction ability of this model is comparable to the supervised learning model. Therefore, SimCLR is introduced to train a better model to extract patch features. Otherwise, the Transformer has been widely used for sequence problems recently. The Transformer includes position encodings and self-attention modules. Through certain position encodings, the model can easily restore the WSI spatial information. For each unit in the input sequence, self-attention can get the weight of attention from other units. The weight of attention reflects the correlation between the patches. Since it is impossible to input a whole WSI to fuse the spatial information, we extract patch features selected from the WSI. The features are added with corresponding position encodings to form a sequence to input into the Transformer. The self-attention in Transformer can automatically learn the correlation between patches, and spatial information is also learned through position encodings to obtain patient-level features.

In this paper, we propose a model called SeTranSurv, which adopts **Se**lf-Supervised learning (SSL) to obtain accurate features from WSIs and employs **Tran**sformer to aggregate patches according to their correlation and spatial distribution. The contributions are summarized as follows: (1) We adopt SimCLR to get a better representation of patch features; (2) We employ the Transformer

to aggregate patches according to their correlation and spatial distribution; (3) We use attentional mechanisms to automatically locate features that are highly relevant to survival analysis, which provides better interpretability. Our work attempt to construct the spatial information and correlation between patches and integrates them for accurate survival prediction in WSIs. Extensive experiments on WSI datasets demonstrate that our model outperforms the state-of-the-art models by providing more precise survival risk predictions.

2 Methodology

An overview of the proposed framework is shown in Fig. 1. Motivated by WSISA [22]: Firstly, we select patches from the non-background area of each WSI and use all of them to train the SSL model by SimCLR [1] to train a feature extraction model ResNet18 [7]. Secondly, we re-select 600 patches in the non-background area of each WSI and then use the ResNet18 trained by SimCLR to extract features for every patch. At the third step, a Transformer Encoder takes both each patch feature and the corresponding position embedding information as input. Finally, The fused information from Transformer Encoder is sent to a Muti-Layer Perception (MLP) to get the final risk score.

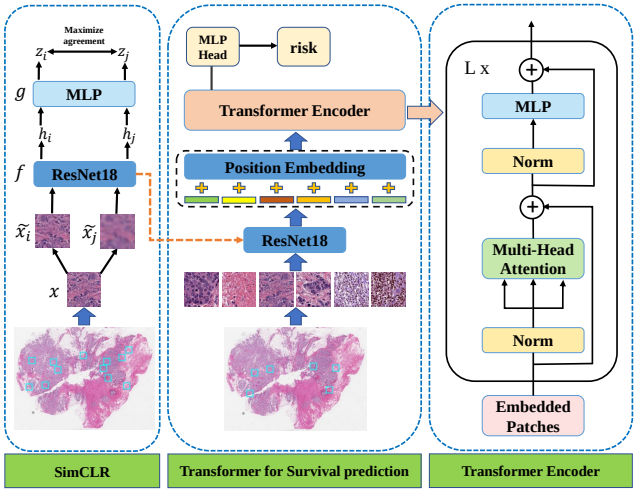

Fig. 1. An overview of the proposed framework. The left part is the advanced SSL method SimCLR [1] for training a feature extraction model ResNet18 [7]. Besides, the middle section is the flow of Transformer Encoder [5] for survival analysis and the right section is the detail of the Transformer Encoder block.

Sampling from WSIs. The primary purpose of this stage is to select some patches from WSIs. A patient often has multiple WSIs, and patch candidates

from different WSIs of the patient reflect the survival risk collectively. Therefore, we extract patches from non-background area of each WSI of the same patient and aggregate their WSI results later. The patches with the size of $512 \times 512 \times 3$ are extracted from 20X (0.5 microns per pixel) to capture detailed information of the images. While randomly selecting the patches, we also record the corresponding coordinate values of the patches in the original WSI image to facilitate the subsequent position encoding.

Train Feature Extraction Model via Self-Supervised Learning. The main goal of this step is to obtain a model that can extract patch features better than the ImageNet pre-trained model. There is a big difference between the ImageNet and WSIs, and we do not have corresponding patch labels to fine-tune the model. SimCLR is a self-supervised learning model proposed by Hinton [1]. SimCLR can train a model with excellent feature extraction ability without the use of labels, and the feature extraction ability of the trained model is comparable to supervised learning. For WSIs, which are significantly different from natural images, we select many unlabeled patches from the WSIs in the training set. These patches are used to train a feature extraction model, ResNet18, through SimCLR. The trained ResNet18 is used to extract the characteristics of patches in the following work.

The specific workflow of SimCLR is shown on the left of Fig. 1. SimCLR learns representations by maximizing consistency between differently augmented views of the same data example. The way to maximizing the consistency is to use the contrastive loss [6] in the potential space. This framework comprises the following four major components: (1) A data augmentation module that transforms any given data example randomly to two correlated views of the same example. The data augmentation followed the same strategies as used in SimCLR for natural images. The image is represented by x. We use two different data augmentation methods to get $\widetilde{x}_i$ and $\widetilde{x}_j$, which is regarded as a position pair. (2) A neural network based encoder f that extracts representation vectors from augmented data examples. We use ResNet18 [7] to obtain patch features $h_i = f(\widetilde{x}_i)$ where h_i is the output after the average pooling layer. (3) A small MLP g maps representations to the space where contrastive loss is applied. We use MLP to obtain $z_i = g(h_i) = W^{(2)}ReLU(W^{(1)}h_i)$. (4) A contrastive loss function defined for a contrastive prediction task. For a given batch size N, the set $\{\widetilde{x}_k\}, k \in \{0...N\}$ includes a positive pair of examples $\widetilde{x}_i$ and $\widetilde{x}_j$. The contrastive prediction task aims to identify $\widetilde{x}_j$ in $\{\widetilde{x}_k\}_{k \neq i}$ for a given $\widetilde{x}_i$.

We randomly sample a minibatch of N examples and define the contrastive prediction task on pairs of augmented examples derived from the minibatch, resulting in $2N$ data points. We do not sample negative examples explicitly. Instead, given a positive pair, similar to [2], we treat the other $2(N-1)$ augmented examples within a minibatch as negative examples. We use NT-Xent loss [12] to optimize the model to enhance feature extraction ability for the ResNet18. These steps ensure that the views of different images are far apart in the potential space and the views of the same image are close together, thus improving the

model's presentation capability. Therefore, we can train a model with excellent feature extraction ability through contrastive learning without labels. The model avoids the inapplicability of features caused by differences of data in different fields compared with the ImageNet pre-trained model.

Feature Fusion via Transformer with Position Encoding. The Transformer includes position encodings and self-attention modules. Through certain position encodings, the model can easily restore the WSI spatial information. For each unit in the input sequence, self-attention can get the weight of attention from other units. The weight of attention reflects the correlation between the patches. Our Transformer Encoder [5] for WSIs follows the architecture design for NLP [5]. For a whole slide image, we sample N (set as 600) patches to get $\boldsymbol{X} \in \mathbf{R}^{N \times H \times W \times C}$ as input into the ResNet18 trained by SimCLR, and the output $\boldsymbol{h} = f(\boldsymbol{X}) \in \mathbf{R}^{N \times 512}$ represents the feature of N patches. $(H \times W \times C)$ is the shape of patches corresponding to $(512 \times 512 \times 3)$.

Position embeddings [5, 15] are added to the patch embeddings to retain position information. We use two-dimensional positional emebedding [5] in this work. To be specific, consider the inputs as a grid of patches in two dimensions. The corresponding horizontal coordinate and vertical coordinate of each patch in WSI are embedding respectively to obtain the position encodings. The x axis and y axis are represented by X-embedding, and Y-embedding, respectively. The embedding size of x axis and y axis are both 24. The 48-dimensional position vector $\boldsymbol{p} \in \mathbf{R}^{N \times 48}$ is spliced with the corresponding vector $\boldsymbol{h}$ to form a 560-dimensional feature vector, $\boldsymbol{z_0} = \boldsymbol{h} \oplus \boldsymbol{p}$, where $\oplus$ is the concatenation operator and $\boldsymbol{z}_0 \in \mathbf{R}^{N \times 560}$. We input $\boldsymbol{z_0}$ into the Transformer Encoder for the integration of features and spatial information.

As shown on the right of Fig. 1, the Transformer Encoder [5] is composed of multiple encoding blocks, and every encoding block has constant widths. The encoding block dimension N is the same as the number of patches sampled from a WSI. Similar to the token of BERT [3], we prepend a learnable embedding to the sequence of embedded patches ($\boldsymbol{z}_0^0$), whose state at the output of the Transformer encoder ($\boldsymbol{z}_L^0$) serves as the WSI representation y. The Transformer encoder consists of alternating layers of multiheaded self-attention [3] (MSA) and MLP blocks (Eq. (1), (2)). The self-attention module can calculate the correlation between the features of different patches through attention mechanism. Layernorm (LN) is applied before every block and residual connections [7] after every block. The MLP contains two layers with a GELU non-linearity. Our Transformer Encoder is composed of six encoding blocks, and each encoding block has four heads, among which the hidden size of MLP is 128.

$$\boldsymbol{t_l} = MSA(LN(\boldsymbol{z_{l-1}})) + \boldsymbol{z_{l-1}}, \qquad l = 1...L \tag{1}$$

$$\boldsymbol{z}_l = MLP(LN(\boldsymbol{t_l})) + \boldsymbol{t_l}, \qquad l = 1...L \tag{2}$$

$$L(\boldsymbol{R}) = \sum_{i \in \{i:S_i=1\}} (-R_i + log \sum_{j \in \{j:T_j \leq T_i\}} exp(R_j)) \tag{3}$$

The output $y = LN(\boldsymbol{z_L^0})$ represents the high-level semantic features fusion by Transformer Encoder. It goes through an MLP Head module [5] $R = W^{(2)}ReLU(W^{(1)}y)$ and directly generates predicted risks (Eq. 3). We integrate the regression of survival risk with high-level feature learning on WSIs. For a patient with multiple WSIs, we average the risk scores of all WSIs for the patient and get the final risk score. The loss function [22] is negative Cox log partial likelihood (Eq. 3) for censored survival data, and S_i, T_i are the censoring status and the survival time of $i-th$ patient, respectively.

3 Experiments

Dataset Description and Baselines. To verify the validity and generalization of SeTranSurv, we apply our methods on three different-sized cancer survival datasets with whole slide pathological images. They are collected from TCGA [8]. The three datasets are Ovarian serous cystadenocarcinoma (OV), Lung squamous cell carcinoma (LUSC), and Breast invasive carcinoma (BRCA). The OV, LUSC and BRCA correspond to small, medium, and large datasets, respectively. The datasets are prepared for multi-omics study, we keep samples with complete multi-omics data. Some statistic facts of WSIs used in the experiments are listed in Table 1. We perform a 5 fold cross-validation on all these datasets.

Table 1. Dataset Statistics. Some patients may have multiple WSIs on record.

Cancer Subtype	No. Patients	No. Censored	No. WSIs	No. Valid patches
LUSC	329	194	512	117649
OV	298	120	1481	196302
BRCA	609	530	1316	274600

SeTranSurv achieves survival analysis from WSIs without using RoIs annotations, so we compare it with the state of the art methods in survival prediction of WSIs without using RoIs annotations, including WSISA [22], DeepGraphSurv [11], CapSurv [14], DeepAttnMISL [19] and RankSurv [4].

Implementation Details. We use Adam optimizer to optimize all methods, and the learning rate is set to 3e-4 by default. We only changed the batch size to 512 for a balance of performance and running time in SimCLR, and the rest of the parameters are the same as SimCLR. We train the Transformer part with a mini-batch size of 32. All hyperparameters were determined for optimal performance on the validation set. Experiments are conducted on a single NVIDIA GeForce GTX 1080 GPU with 11 GB memory.

Results and Discussions. To assess the performance of SeTranSurv, we use the concordance index (C-index) as the evaluation metric. C-index is a standard evaluation metric in survival prediction [9]. It ranges from 0 to 1. The larger the C-index is, the better the model predicts. The training time of our model increases linearly with the sample size. It takes about 38 hours for the SimCLR to train a ResNet18 through WSI patches, and about 6 hours to train the final Transformer model for LUSC dataset.

Table 2. Performance comparison of the proposed method and other methods using C-index values on three datasets. The method that using the SimCLR to extract the patch features on the basis of the original method is marked with * in the table. We use OursV1 to indicate that SimCLR and position information are not used in our method, and use OursV2 to indicate that SimCLR is not used in our method.

Model	LUSC	OV	BRCA
WSISA [22]	0.612	0.601	**0.637**
WSISA *	**0.636**	**0.610**	0.639
DeepGraphSurv [11]	0.647	0.640	0.674
DeepGraphSurv *	**0.675**	**0.659**	**0.685**
CapSurv [14]	0.660	0.641	0.662
CapSurv *	**0.665**	**0.653**	**0.671**
DeepAttnMISL [19]	0.670	0.659	0.675
RankSurv [4]	0.674	0.667	0.687
OursV1	0.662	0.655	0.686
OursV2	0.687	0.673	0.690
SeTranSurv	**0.701**	**0.692**	**0.705**

Table 2 shows the C-index values on three datasets. Our method attains the best C-index values that present the best prediction performance among all methods. Our approach outperforms the previous best approach by an average of 3% on all three datasets. The result illustrates the proposed method is effective and universal.

To explore the effectiveness of SimCLR in extracting features from patches, we conduct a comparative experiment in all methods on whether use the model trained by SimCLR to extract features of patches. As can be seen from Table 2, the features extracted from the SimCLR-trained model can improve the results well in almost methods, which indicates that the model trained with SSL in WSI patches can obtain a better feature extraction ability than the ImageNet pre-trained model. To verify the effectiveness of self-attention and location information in our method, we conduct an ablation experiment. The result of OursV1 shows that self-attention can learn the correlation between patches and obtain good results. The result of OursV2 indicate that position information enables the model to combine the spatial information of entire WSI, which also improves the results. The above results indicate SeTranSurv has better feature extraction ability and feature aggregation capability.

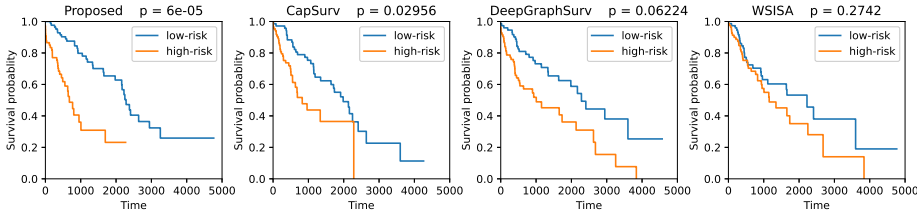

Fig. 2. Kaplan-Meier survival curves of different methods for LUSC datasets in the test set. High risk (higher than the median) groups are plotted as brown lines, and low risk (lower than or equal to median) groups are plotted as blue lines. The x-axis shows the time in days, and the y-axis presents the survival probability. Log-rank p-value is displayed on each figure.

Given the trained survival models, we can classify patients into low-risk or high-risk groups for personalized treatments by the predicted risk scores in the test set. Two groups are classified by the median of the predicted risk score. Patients with longer survival time should be divided into the low-risk group, and with short survival time should be divide into the high-risk group. To measure if those models can correctly divide patients into two groups, we draw Kaplan-Meier survival curves of LUSC dataset in Fig. 2. The log-rank test is conducted to test the difference between two curves and evaluate how well the model will classify testing patients into low and high-risk groups. It is shown that the proposed method can attain the most significant result of the log-rank test.

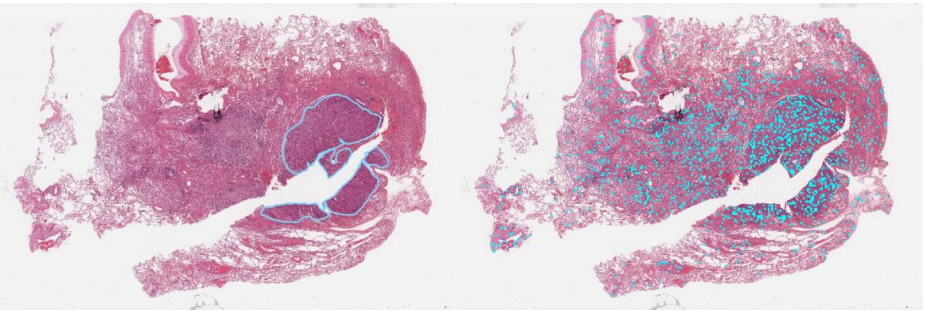

Fig. 3. Left: annotation of RoIs; Right: The blue part represents the parts with larger weight given by the model in the randomly selected patches. It can be seen that the patches selected in the RoIs region are generally given a relatively large weight, while only a small number of patches in the non-RoIs region are given a relatively large weight.

SeTranSurv uses the attention mechanism to recognize significant patterns in WSI automatically. As shown in Fig. 3, we draw all the randomly selected

patches whose weight assigned by the model exceed the median. We measured the learned attentions patches in Fig. 3, and 20 repeated experiments showed that 85% of selected patches had greater attention values than the average in the ROI regions, significantly higher than the 26% in non-ROI regions. The result indicates that the model can identify the patches that are highly correlated with survival analysis and give these patches a large weight. The patches with large weight correctly highlight most of the RoIs annotated by medical experts, which shows that our method can locate useful features and have good interpretability.

4 Conclusion

We propose SeTranSurv to combine SSL and the Transformer for survival analysis in WSIs. SeTranSurv can extract patch features better and use the correlation and position information between patches to fuse the features that are useful for survival analysis in the entire WSI. Extensive experiments on three large cancer datasets indicate the effectiveness of SeTranSurv.

Acknowledgements. This work was supported by the Meizhou Major Scientific and Technological Innovation Platforms and Projects of Guangdong Provincial Science&Technology Plan Projects under Grant No. 2019A0102005.

References

1. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: ICML 2020: 37th International Conference on Machine Learning. vol. 1, pp. 1597–1607 (2020)
2. Chen, T., Sun, Y., Shi, Y., Hong, L.: On sampling strategies for neural network-based collaborative filtering. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 767–776 (2017)
3. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.N.: Bert: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). pp. 4171–4186 (2018)
4. Di, D., Li, S., Zhang, J., Gao, Y.: Ranking-based survival prediction on histopathological whole-slide images. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 428–438 (2020)
5. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations (2020)
6. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 9729–9738 (2020)
7. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 770–778 (2016)

8. Kandoth, C., McLellan, M.D., Vandin, F., Ye, K., Niu, B., Lu, C., Xie, M., Zhang, Q., McMichael, J.F., Wyczalkowski, M.A., Leiserson, M.D.M., Miller, C.A., Welch, J.S., Walter, M.J., Wendl, M.C., Ley, T.J., Wilson, R.K., Raphael, B.J., Ding, L.: Mutational landscape and significance across 12 major cancer types. Nature **502**(7471), 333–339 (2013)
9. Katzman, J.L., Shaham, U., Cloninger, A., Bates, J., Jiang, T., Kluger, Y.: Deepsurv: personalized treatment recommender system using a cox proportional hazards deep neural network. BMC Medical Research Methodology **18**(1), 24–24 (2018)
10. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. In: ICLR (Poster) (2016)
11. Li, R., Yao, J., Zhu, X., Li, Y., Huang, J.: Graph cnn for survival analysis on whole slide pathological images. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 174–182 (2018)
12. van den Oord, A., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748 (2018)
13. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: ICLR 2015 : International Conference on Learning Representations 2015 (2015)
14. Tang, B., Li, A., Li, B., Wang, M.: Capsurv: Capsule network for survival analysis with whole slide pathological images. IEEE Access **7**, 26022–26030 (2019)
15. Wang, B., Zhao, D., Lioma, C., Li, Q., Zhang, P., Simonsen, J.G.: Encoding word order in complex embeddings. In: ICLR 2020 : Eighth International Conference on Learning Representations (2020)
16. Wang, H., Xing, F., Su, H., Stromberg, A.J., Yang, L.: Novel image markers for non-small cell lung cancer classification and survival prediction. BMC Bioinformatics **15**(1), 310–310 (2014)
17. Wang, S., Yao, J., Xu, Z., Huang, J.: Subtype cell detection with an accelerated deep convolution neural network. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 640–648 (2016)
18. Yao, J., Wang, S., Zhu, X., Huang, J.: Imaging biomarker discovery for lung cancer survival prediction. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 649–657 (2016)
19. Yao, J., Zhu, X., Jonnagaddala, J., Hawkins, N.J., Huang, J.: Whole slide images based cancer survival prediction using attention guided deep multiple instance learning networks. Medical Image Analysis **65**, 101789 (2020)
20. Yu, K.H., Zhang, C., Berry, G.J., Altman, R.B., R, C., Rubin, D.L., Snyder, M.: Predicting non-small cell lung cancer prognosis by fully automated microscopic pathology image features. Nature Communications **7**(1), 12474–12474 (2016)
21. Zhu, X., Yao, J., Huang, J.: Deep convolutional neural network for survival analysis with pathological images. In: 2016 IEEE International Conference on Bioinformatics and Biomedicine (BIBM). pp. 544–547 (2016)
22. Zhu, X., Yao, J., Zhu, F., Huang, J.: Wsisa: Making survival prediction from whole slide histopathological images. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6855–6863 (2017)