Cancer Cell Cluster Detection by Unified Tumor Tissue Slide Image and Gene Expression Quantification

Wenkang Zhan, Zhengkang Fan

Introduction

Cancer is one of the most fatal diseases of human beings. Nowadays, diagnosis of cancer is usually based on a tissue slide from a site. One problem is that cancer cells may transfer from one site to another site, which is called metastasis. It is difficult to distinguish a cancer metastasis region in tumor slides. Building a machine learning model which can identify the possible cancer cell cluster region could ease the effort of diagnosis. As deep learning algorithms, especially convolutional neural networks, have achieved great success in the computer vision field, it should also be functional to medical images. Slide images are widely provided by multiple data providers, including The Cancer Genome Atlas (TCGA). By having a thought that the tumor slides have the potential to distinguish metastasis from sites, we plan to use AI techniques on processing images to identify cell clusters from primary tumor sites from cell clusters from other sites. However, the slide image dataset is usually not large, so we cannot get enough data for a complicated network. Another powerful material to classify different types of tumors is gene expression material. Employing gene expression as a heuristic information, we want to combine the gene expression and the tissue slides together. In this project, we plan to unify the gene expression data with the slide images by a reasoning system to improve the accuracy of predicting cancer, and build a machine learning model to mark the cancer region in slide images.

Related works

As deep learning is rapidly developed on computer vision field, these neural network methods have been applied to objectively evaluate high-dimensional medical data and high-resolution images, such as such as computed tomography (CT), positron emission tomography (PET) and magnetic resonance imaging (MRI) data. Rather than being chosen by human selection, machine learning and statistical methods also enable image features to be identified by algorithms. A deep convolutional neural network (CNN) is employed to predict future cognitive decline in Alzheimer’s disease based on flurodeoxyglucose and florbetapir positron emission tomography (PET) [6]. A convolutional neural network with maximum-likelihood is employed to reconstruct the PET/MRI by a CT-derived attenuation map [7].

Gene expression data has been applied to functional genomics search. Machine learning methods have been applied to medical data and have shown remarkable performance in differential diagnosis. A deep learning model indicates some inner pattern of gene data that the entire gene regulatory structure and specific combination of regulatory elements define gene expression levels [8]. Based on the feature selection method, a support vector machine is employed to identify and analyze blood gene expression for Osteoarthritis [9].

Detection of Cancer Cell Cluster in slide images could be viewed as an imaging biomarker problem, which is to biologic features relevant to a patient’s diagnosis [4].

A number of biomarkers are frequently used to determine risk of cancer or other diseases. Interpretable machine learning with biomarker cues is employed for melanoma detection [5]. Random forest, support vector machine, naive bayes classifiers identify the gene data, predict the survival of breast cancer and discover the potential biomarkers which are strongly related to breast cancer survivability and cancer in general.

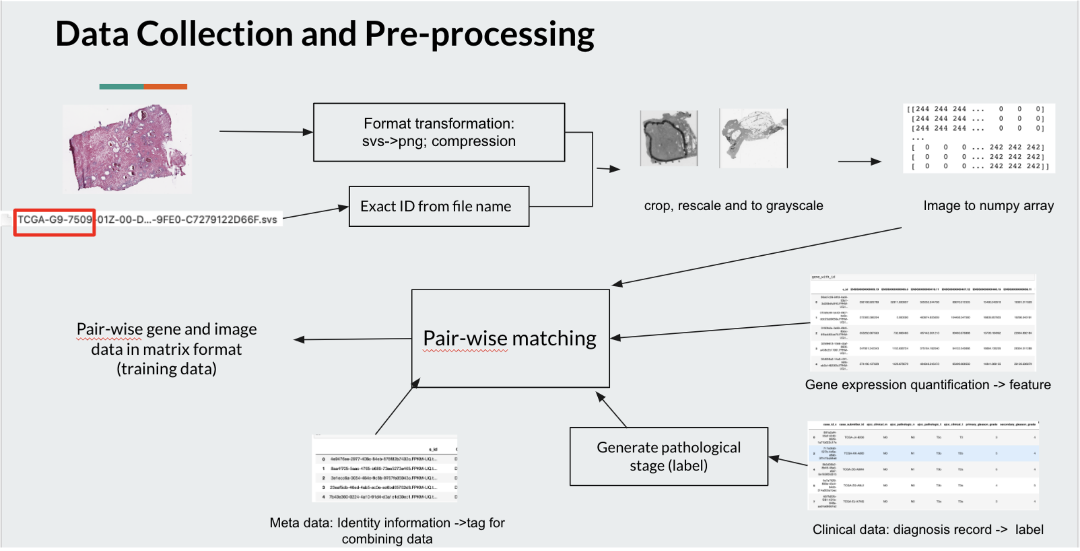
Data collection

The data of this project is collected from The Cancer Genome Atlas (TCGA) data portal, which provides different kinds of data, such as tissue slide image samples, gene expression quantification, clinical data, RNA sequences, etc., of over 200 cancer types.

In this project, both the tissue slide images and the gene expression quantification data are collected from The Cancer Genome Atlas (TCGA) data portal [1]. The slide image data is collected from the category 'slide image->prostate gland'. The quantification data is collected from the category 'gene expression quantification-> HTSeq - FPKM'. From the category 'gene expression quantification-> HTSeq - FPKM', we can also download the corresponding clinical data (includes diagnosis records which may become the labels), meta data (identity tags for data combination).

Data preprocessing

The data provided in the TCGA data portal is raw data. We need to process the raw data and transform them into matrix format, as machine learning models take matrices as inputs. Generally, the preprocessing part can be divided into three parts: (1) slide image processing, (2) feature reduction for gene expression quantification, and (3) pairwise matching. The overview of the data preprocessing is shown as Figure 1.



Data Collection and Pre-processing

Format transformation: svs->png; compression

Exact ID from file name

crop, rescale and to grayscale

Image to numpy array

```
[[244 244 244 ...   0   0   0]
 [244 244 244 ...   0   0   0]
 [244 244 244 ...   0   0   0]
 ...
 [  0   0   0 ... 242 242 242]
 [  0   0   0 ... 242 242 242]
 [  0   0   0 ... 242 242 242]]
```

Pair-wise matching

Pair-wise gene and image data in matrix format (training data)

Gene expression quantification -> feature

Generate pathological stage (label)

Meta data: Identity information ->tag for combining data

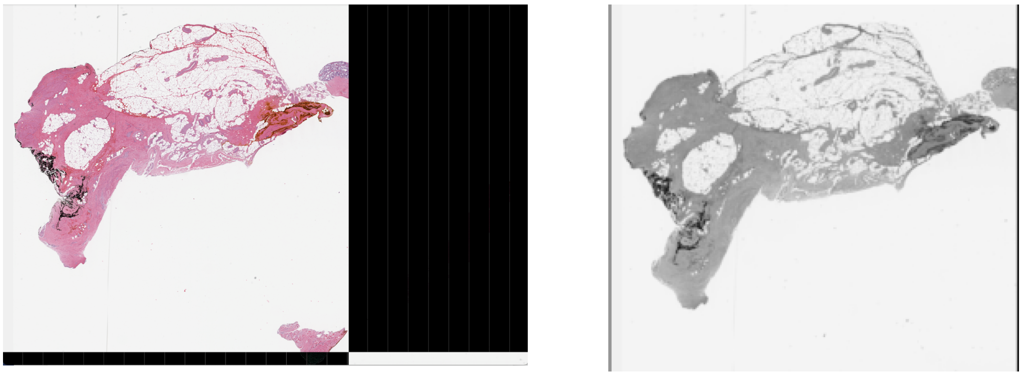
Clinical data: diagnosis record -> label

Figure 1. Data preprocessing

Slide image preprocessing

As the downloaded data is not well organized as a benchmark dataset, data preprocessing is needed before the downloaded data sets are fed to machine learning/deep learning models. Slide images are very large, with an average width of 101,688 pixels and an average height of 73,154 pixels and each slide image has different shapes. If these images are directly used to model training, it would be prohibitively computationally expensive. Also, the original images contain useless information such as background, shadows, water, smudges, and pen marks. Thus, we need to scale down the slide images, select the informative portions, and crop them into a fixed size [2]. Finally, as we use python with machine learning toolboxes to realize our project, we need to further transform the processed images into matrix format, i.e. numpy arrays.

We use packages in the python environment such as OpenSlide, PIL, pandas, numpy, glob, etc. to process the slide image. First, we use glob package to traverse each image in the directory, and extract the unique IDs from their file names with string comparison technique. Then, we use the OpenSlide package to transform the slide images from svs format into PIL image format. After we get the slide images of PIL format, we use PIL package to crop out the main region of the tumor in the slide image, transform the RGB images into grayscale images (due to computational resources limitation) and resize the image into 256*256 pixels. Finally, we transform the PIL images into numpy arrays, and combine each image with its ID as a list. We also keep copies of processed slide images as png format. Figure 2 shows the results of slide image processing.



(a) Before processing

(b) After processing

Figure 2. An example of slide image processing

Apart from the slide image preprocessing proposed above, we also need to find out an unique key to combine the slide image with the gene expression quantification and the label. In this project, unique IDs of each slide image corresponding to the gene expression quantification and label are stored in the file name of the slide image. We use string comparison techniques to extract the file IDs from the file names.

Gene expression quantification (features) and label

As for the gene expression quantification data, the gene data (features) and clinic data with diagnosis (labels) are in separated data files, we need to merge them into a grid form matrix by provided ID number. The gene expression quantification is fat data, with over 60,000 features but only hundreds of samples. Therefore, feature selection must be performed before data is fed to machine learning/deep learning models.

In our project on cancer cluster detection, the TCGA data portal provides gene expression quantification, clinical data, meta data which contain the unique information of each patient, and supplemental clinical data which contains all of the unprocessed data related to the patient. The gene expression quantification can be treated as features, and we can find the corresponding labels in the clinical data. We can merge the features and label pair-wisely with the help of the meta data file, which contains the identity information such as sample IDs, file IDs, etc.

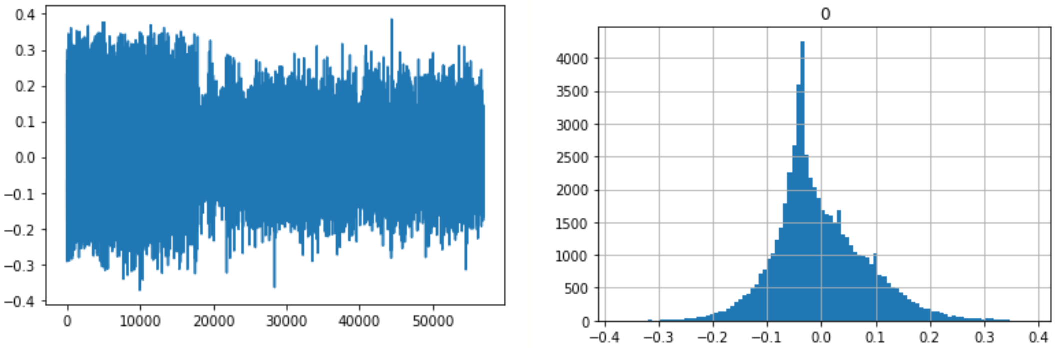
In our project, we use pathological state as our label, which isn’t not directly provided in the clinical data file. We need to generate the pathological stage according to the diagnosis records in the clinical data file based on the American Joint Committee on Cancer (AJCC) TNM staging system. We can generate the pathological stages according to the American Joint Committee on Cancer (AJCC) TNM staging system. The TNM system for prostate cancer is based on five information shown as belows.
1.The extent of the main (primary) tumor (T category).
2. Whether the cancer has spread to lymph nodes (N category).
3. Whether the cancer has metastasized to other parts of the body (M category).
4. The prostate-specific antigen (PSA) level at the time of diagnosis.
5. The Gleason score. It measures how likely the cancer is to grow and spread.

Feature reduction

In gene expression quantification, the number of features (over 60,000) is far more than the number of samples (hundreds of samples), which is called fat data. If we feed fat data directly into a machine learning model, it may lead to many problems such as computational resources problems, overfitting etc. In order to avoid or eliminate these problems, we’d better reduce the feature dimension before we train a machine learning model. In the project, we use two methods for feature reduction, one is a machine learning method, and the other is based on domain knowledge.

(a)Machine learning method

We use correlation between each feature and label as an indicator to reduce the dimension of features.. We set a threshold, and select those features having a higher correlation than the threshold. In this process, we used a wrapper scheme. We trained baseline models with the selected features by different thresholds, and compared their performance. We use the performances of baselines and the number of remaining features as an evaluation of our feature reduction. The correlation between features and the label shown as Figure 3. We set the threshold to {0.16, 0.18 ,0.20, 0.22, 0.24, 0.26, 0.28, 0.30}, and compare their performances on baselines.



(a) Correlation

(b) Distribution of correlation

Figure 3. Correlation between features and label

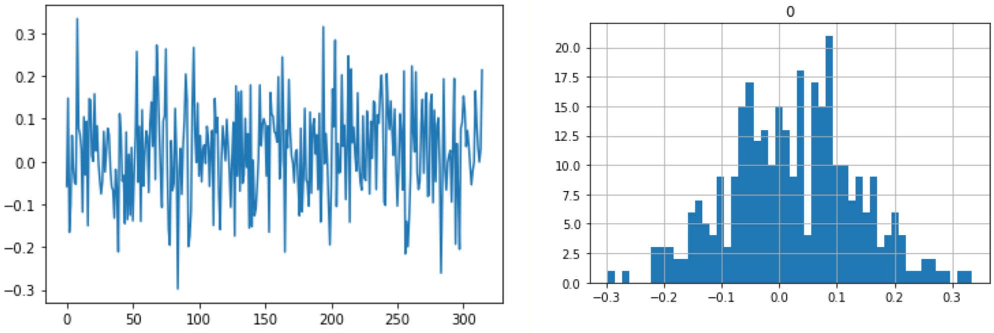
Table 1. Number of remaining features with different thresholds

threshold	0.16	0.18	0.2	0.22	0.24	0.26	0.28	0.30
#features	3566	2333	1534	969	623	397	238	132

(b)Domain knowledge method

We searched for the gene which played a critical role in the pathological prostate stage, and manually selected those genes from the whole features. Many factors increase a person’s chance of developing prostate cancer. Factors including age, race, location, family history, genetic mutations, exposure, diet etc., raise the risk of developing prostate cancer. In this project, we consider the gene related to prostate cancer. The TCGA data portal provides a list of genes which may have a closer relationship with prostate cancer, such as ALK, SS18L1, RABEP1, etc. We downloaded this list of gene positions corresponding to the gene expression quantification, and screened out these listed features from the original gene expression quantification. We analyze the correlations of each selected gene with the prostate pathological stage, and use these selected features to train baseline models.

Figure 3 is the correlations between each feature selected by domain knowledge and the label. From Figure 3, we surprisingly see that the features selected by domain knowledge are not necessarily highly related to the label, which displays a different result from that of feature reduction by machine learning method.



(a) Correlation

(b) Distribution of correlation

Figure 3. Correlation between selected features by domain knowledge and label

Table 3. Accuracy of baselines

	Selected features by correlations								Domain knowledge
thres.	0.16	0.18	0.20	0.22	0.24	0.26	0.28	0.30	-
#feat.	3566	2333	1534	969	623	397	238	132	315
LR	**0.66**	0.66	0.6	0.63	0.63	0.6	0.56	0.56	**0.66**
CNN	0.44	0.43	0.43	0.48	0.48	**0.56**	**0.56**	**0.56**	**0.56**
LSTM	0.46	0.48	0.48	0.48	0.48	0.53	**0.56**	**0.56**	**0.56**

Pairwise matching

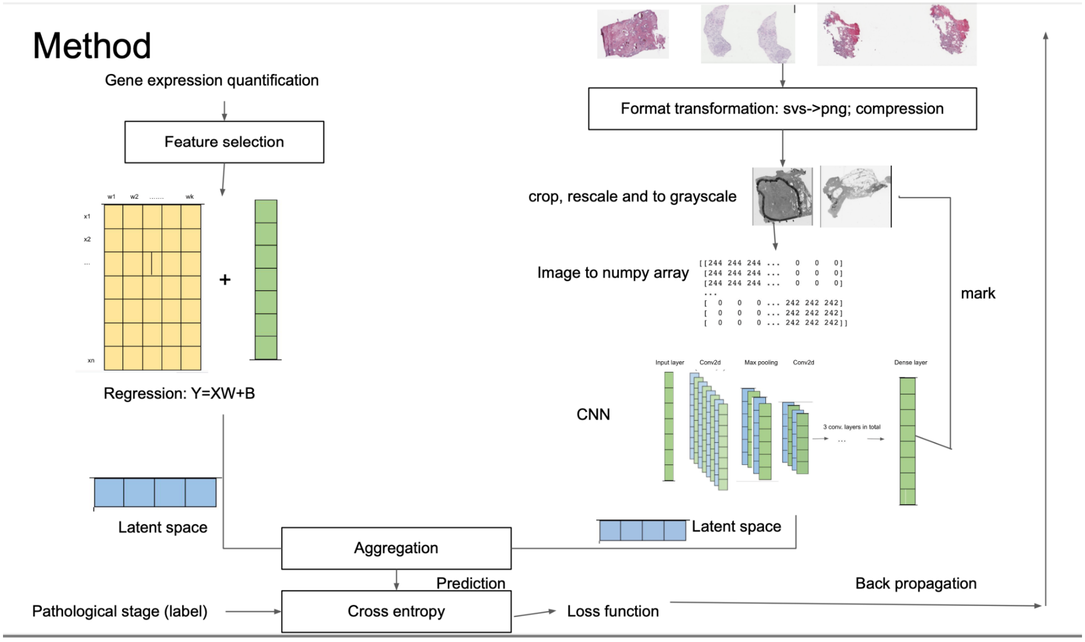
After the preprocessing on the slide images and gene expression quantification, we use meta data which contains identity information such as sample IDs, file IDs etc., to merge the gene expression quantification, slide images, and pathological stages (labels) pairwisely together into numpy arrays.

Normalization

For gene expression quantification, we perform min max normalization to each column of the features and rescaled them to a range of [0,1]. For slide images, each pixel is divided by 255, and is rescaled to [0,1].

Method

This project aims to mark the cancer cell cluster regions which are related to tumor in the tumor tissue slide images by a machine learning/deep learning system. As the TCGA dataset provides pair-wise image and gene expression data, we combine both the slide images and the gene expression quantification data to train a machine learning system to predict the risk of tumor and then use the trained models to mark cancer clusters in slide images. The overview of our method is shown as Figure 4.



```
[[244 244 244 ...   0   0   0]
 [244 244 244 ...   0   0   0]
 [244 244 244 ...   0   0   0]
 ...
 [  0   0   0 ... 242 242 242]
 [  0   0   0 ... 242 242 242]
 [  0   0   0 ... 242 242 242]]
```

Figure 4. Overview of our method

This machine learning system mainly has three parts, the first is a machine learning/deep learning algorithm trained by gene expression quantification data, the second part is a convolutional neural network trained on slide images. The final part is the neural network, which combines the outputs of the two trained models to make a prediction on the risk of tumor. Based on the trained convolutional neural network, we can map the intermediate variables to the original images to mark the cancer region in the slide image. As you can see in Figure 5, here is our model structure.

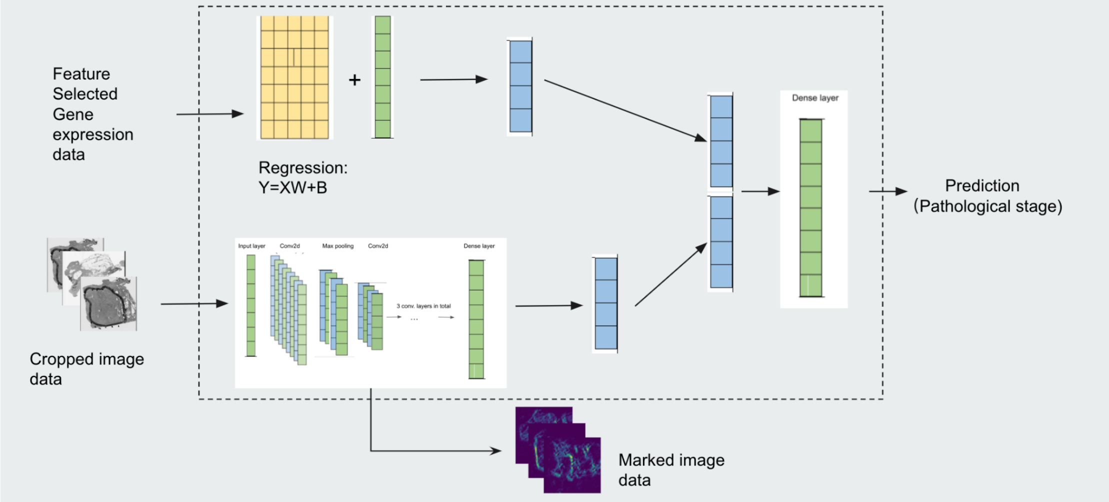


Figure 5 . Model structure

(1) **Image modeling**: A convolutional neural network (CNN) should be built to capture features from the RGB slide images. As the downloaded images are raw data, data preprocessing is needed before the images are used as input to the CNN. As the raw slide images have different sizes, a python based package OpenSlide is used to downscale the raw slide images, converge them into fixed size. Based on the fix-sized images, a convolutional neural network (CNN) should be employed to learn

the inner patterns of the images. We built a three layer convolutional neural network with max pooling layer, which helps us to get the higher level information from the image.

(2) **Genomic data modeling**: Information should be extracted from the gene expression quantification data. In the downloaded file, gene expression data (features) and clinic data (labels) are in separated files. They need to be merged by unique ID numbers. As gene data is highly dimensional which has over 60,000 features but only hundreds of samples, feature reduction must be performed before the data is fed into a machine learning/deep learning model. After we do feature selection by domain knowledge, we use 1 dense layer with relu activation function to gain additional information from the Genomic data. Besides, we want to this additional information to improve the performance of prediction of pathological stage.

(3) **Prediction modeling:** From the CNN trained by slide images and the machine learning model trained by gene expression data, we can get two matrices of high-level information. In this step, we combine these two information matrices to improve the prediction of the risk of tumor. We concatenate these two matrices and put them into a 1 layer neural network to make a final prediction on the pathological stage. Based on the dataset we have, we train this model by the back propagation with Adam optimization method.

(4) **Cancer region prediction:** After training, this model could predict the stage of cancer by the image information and gene expression, the CNN based on slide images can capture some inner pattern or high level feature from the image, which could be stored in the hidden units or other intermediate variables. A CNN mapping method should be used to extract information from the intermediate variables, and map them to the original slide images. With this mapping method, we can mark the cancer cell cluster regions in the slide images.

Result

We picked prostate cancer as the target cancer, we downloaded over 200 slide image data from The Cancer Genome Atlas (TCGA), after data preprocessing and matching the gene expression, we got 151 samples and their pathological stages. We split the dataset into 80% training data and 20% validating data. As you can see, after training we get an overfitting model, due to the size of the dataset being too small.

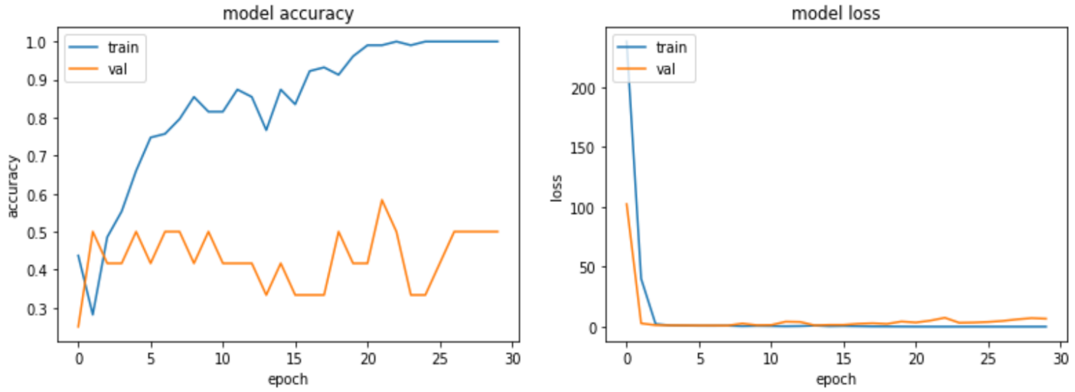


Figure 6. Training model result

However, on the other hand, we can see this model performance very well on the training dataset, which means we could use this model to get the cancer region. So, we developed a mapping method to predict the region on the image. Example as follows.

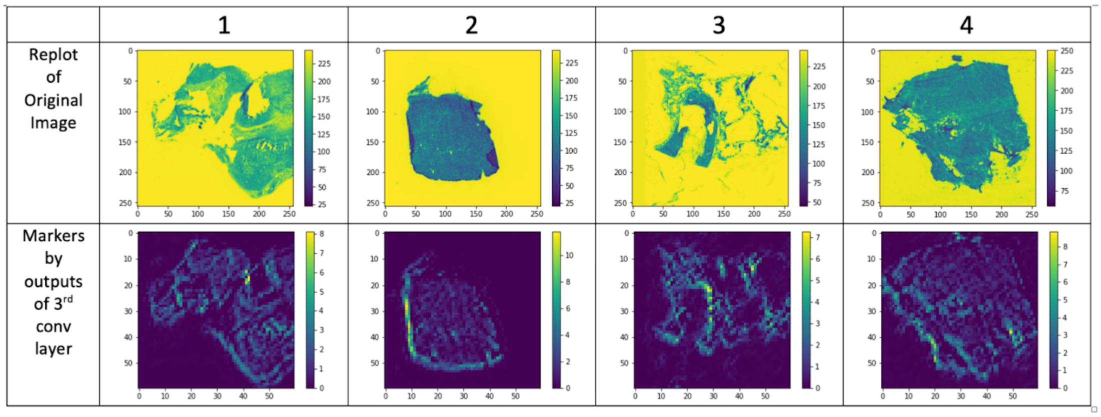


Figure 7. Examples of marked images

Future work

The biggest problem we faced is the insufficient amount of data, even though The Cancer Genome Atlas (TCGA) has more than 2000 samples of prostate cancer, but not all of them have a usable slide image. And the raw data is usually very big, the download time is very long for two people to process it. In the future, since we have time to process more data and feed it into the model, we might get a more robust model. On the other hand, the overfitting issue is there. Since the dataset is not enough and the model is simple enough, we might need to add a dropout layer and regularization term to reduce overfitting. We believe both problems can be solved or improved by adding more data into the model. After we got enough data, we might need to adjust the model structure to create a more powerful model, which can give us better results.

Reference

[1]https://www.cancer.gov/about-nci/organization/ccg/research/structural-genomics/tcga

[2]https://developer.ibm.com/articles/an-automatic-method-to-identify-tissues-from-big-whole-slide-images-pt1/

[3] https://en.wikipedia.org/wiki/Bayesian_network

[4] https://en.wikipedia.org/wiki/Imaging_biomarker

[5] Gareau DS, Browning J, Correa Da Rosa J, et.al. Deep learning-level melanoma detection by interpretable machine learning and imaging biomarker cues. J Biomed Opt. 2020 Nov;25(11):112906. doi: 10.1117/1.JBO.25.11.112906. PMID: 33247560; PMCID: PMC7702097.

[6] Choi H, Jin KH; Alzheimer’s Disease Neuroimaging Initiative. Predicting cognitive decline with deep learning of brain metabolism and amyloid imaging. Behav Brain Res. 2018 May 15;344:103-109. doi: 10.1016/j.bbr.2018.02.017. Epub 2018 Feb 14. PMID: 29454006.

[7] Hwang D, Kang SK, Kim KY, et al. Generation of PET Attenuation Map for Whole-Body Time-of-Flight 18F-FDG PET/MRI Using a Deep Neural Network Trained with Simultaneously Reconstructed Activity and Attenuation Maps. J Nucl Med. 2019 Aug;60(8):1183-1189. doi: 10.2967/jnumed.118.219493. Epub 2019 Jan 25. PMID: 30683763; PMCID: PMC6681691.

[8] Zrimec, J., Börlin, C.S., Buric, F. et al. Deep learning suggests that gene expression is encoded in all parts of a co-evolving interacting gene regulatory structure. Nat Commun 11, 6141 (2020). https://doi.org/10.1038/s41467-020-19921-4

[9] Li J, Lan CN, Kong Y, Feng SS, Huang T. Identification and Analysis of Blood Gene Expression Signature for Osteoarthritis With Advanced Feature Selection Methods. Front Genet. 2018 Aug 30;9:246. doi: 10.3389/fgene.2018.00246. PMID: 30214455; PMCID: PMC6125376.

[10] Wang L, Audenaert P, Michoel T. High-Dimensional Bayesian Network Inference From Systems Genetics Data Using Genetic Node Ordering. Front Genet. 2019 Dec 20;10:1196. doi: 10.3389/fgene.2019.01196. PMID: 31921278; PMCID: PMC6933017.

[11] Agrahari, R., Foroushani, A., Docking, T.R. et al. Applications of Bayesian network models in predicting types of hematological malignancies. Sci Rep 8, 6951 (2018). https://doi.org/10.1038/s41598-018-24758-5