TEAM HMS-MGH-CCDS METHOD1

Aoxiao Zhong, Quanzheng Li
azhong@mgh.harvard.edu, li.quanzheng@mgh.harvard.edu
Harvard Medical School- Mass General Hospital-Center for Clinical Data Science

ABSTRACT

Our algorithm is based on our results of CAMELYON 16.We start with pre-processing for tissue segmentation. Then patches are extracted randomly from tissue region as training samples for a fully convolutional resnet-101 with dilated convolution and atrous spatial pyramid pooling to do segmentation. After training, a tumor probability heatmap is generated for each slide with the trained model. Finally, global and local features are extracted from heatmaps to train a final random forest classifier to discriminate different categories for slides. pN-stage prediction results are directly determined by the rules given.

***Index Terms*—** One, two, three, four, five

1. PRE-PROCESSING

The most part of a whole-slide image is blank. To reduce computations and focus on tissue region, a threshold based segmentation method is adopted to automatically detect tissue region. In particular, the original image from RGB color space is transferred to gray-scale and Otsu algorithm [2] is performed.

2. CANCER METASTASIS DETECTION FRAMEWORK

The whole slide images and the ground truth image annotation are taken as input. We use pixel-wise label for patches instead for dense predictions. The overall framework is depicted in Fig. 1.

Image patches from slides and corresponding pixel-wise ground truth patches are extracted for training the neural networks for segmentation.

After neural networks are trained, sliding window method is used on all the tissue region to generate a tumor probability heatmap for each slide. With this tumor probability heatmap, post-processing is then performed to classify the slides.

Details of training and predicting are showed in following sections.

3. POST-PROCESSING

To get a promising WSI-level classifier, geometrical and morphological features are extracted from tumor probability maps. Thresholding is done on the heatmap to get a binary mask. Then we find the largest (longest main axis) connected region with connectivity of 2. The properties of these region is measured as local features, including the longest axis in the region, ratio of pixels in the region to pixels in the total bounding box, eccentricity of the ellipse that has the same second–moments as the region, the area of the region and the mean intensity of the tumor region. Local features are extracted with several different thresholds. Then global feature like the maximum value for the heatmap is extracted. All these features are combined as a final feature representation vector for each slide. A random forest classifier is trained to discriminate WSIs with different labels.

4. MODEL DETAILS

We use a fully convolutional resnet-101[3] with dilated convolution and atrous spatial pyramid pooling to do segmentation[1].

Conv5 has the dilation of 2. Atrous spatial pyramid pooling has the dilation of 3,6,9,12. The output dense prediction of the model reduce the size of input by a factor of 16.

4. TRAINING DETAILS

To prevent model from over-fitting and save disk space, training patches is extracted randomly on-the-fly. Also data augmentation is applied on patches with random flip, brightness adjustment with max delta of 32, color shift in RGB space with max delta of 10, contrast adjustment with max delta factor of 0.2. The patches are extracted from level 1(20x). The output of the model is on level 5.

We use all the training/testing slides of CAMLYON16 and all the training slides annotated as negative and slides with lesion-level annotated tumor slides in CAMLYON17 training set as lesion-level training set.

To address the problem of training data imbalance that much more normal regions are in the slides then tumor regions. Patches with tumor region (at least one pixel in the patch is labeled as tumor) and without tumor region (no pixels in the patch is labeled as tumor) is extracted uniformly from slides. To extract a patch, we first randomly select one slide which contains this class of patches uniformly. Then we randomly extract patches from the slide uniformly or with weight (for hard example mining). For

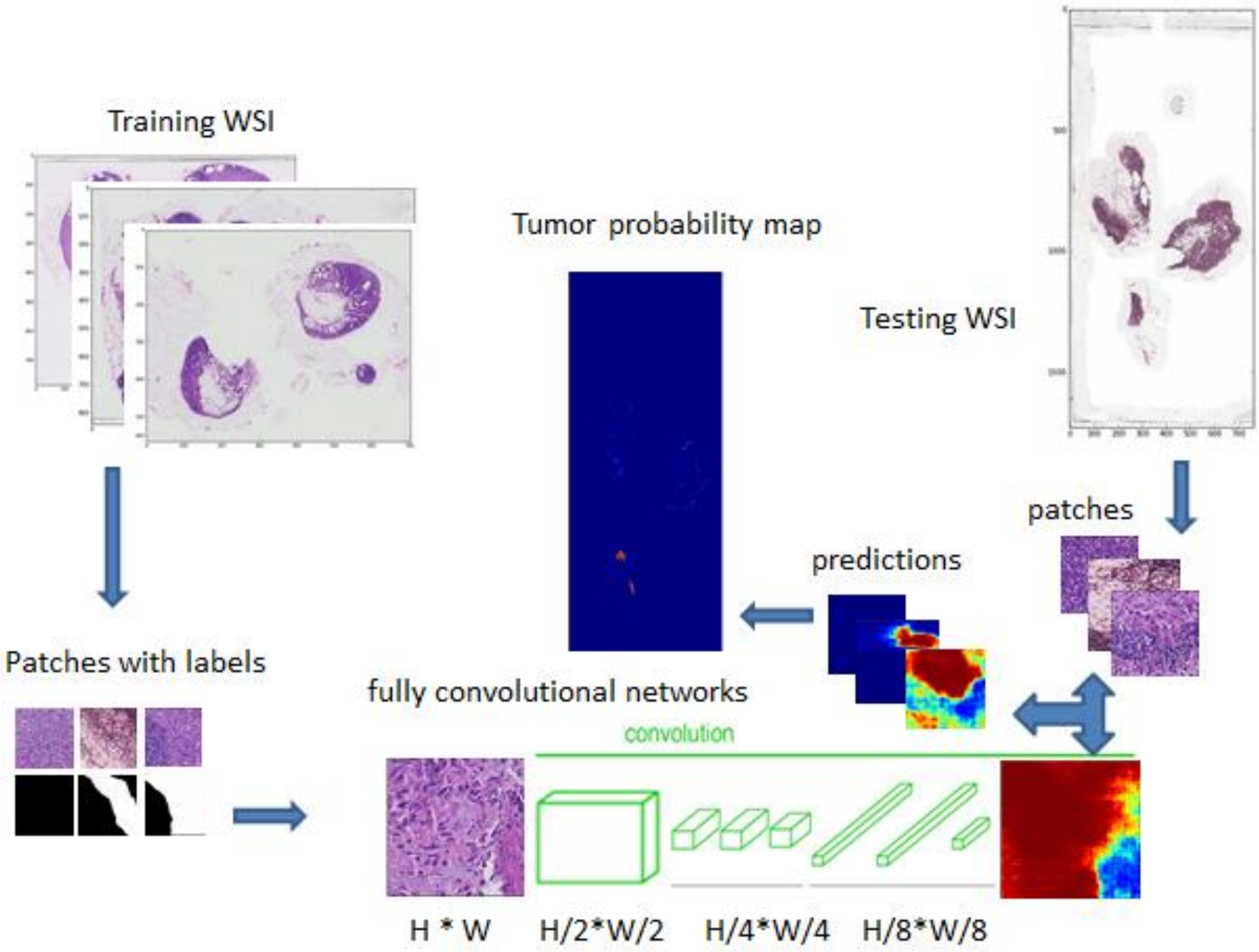

Figure 1 The framework of cancer metastases detection

training, the patch size is 960*960 due to the limitation of GPU's memory.

We use batch size of 10 on each GPU. For each batch, it contains 5 patches with tumor regions and 5 patches without tumor regions. The model initialized with pre-trained model for segmentation task on Microsoft-COCO dataset[4].

.The model is implemented with Caffe[5]. The learning rate starts from 2.5e-4 with a policy of 'poly' (with power = 0.9). We use a weight decay of 0.0005 and a momentum of 0.9. The model is first trained for 10000 iterations on DGX-1(8* Nvidia Tesla P100). Then we run a round of inference on all the training slides to generate a weighted map for hard example mining. With the weighted map, false positives have a higher probability of been chosen as training patches. With hard example mining, we trained another 10000 iterations with fixed learning rate of 1e-4 to get the final model.

5. PREDICTING DETAILS

We use sliding window method on the tissue region the generate tumor probability heatmaps. The input patch size for inference is 1280x1280, but only the central part with size of 1024x1024 is taken as outcomes.

Since the patch is on level 1, the dense prediction map is downsampled 16 times, the output heatmap is on level5.

6. POST-PROCESSING DETAILS

Global features we used:

1.Max value of the heatmap

Local features we used:

For threshold of 0.5,0.9:

For the largest connected region:

1.area
2.extent
3.eccentricity
4.major_axis_length
5.mean_intensity
6.solidity

7. perimeter

For all the connected region:

1.area

We combine all these to get a vector of length 17 to train a random forest classifier to classify slides. pN-stage is directly determined by the rules given.

7. REFERENCES

.

[1] Chen L C, Papandreou G, Kokkinos I, et al. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs[J]. arXiv preprint arXiv:1606.00915, 2016..

[2] Otsu N. A threshold selection method from gray-level histograms[J]. Automatica, 1975, 11(285-296): 23-27.

[3] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 770-778.

[4] Lin T Y, Maire M, Belongie S, et al. Microsoft coco: Common objects in context[C]//European Conference on Computer Vision. Springer International Publishing, 2014: 740-755.

[5] Jia Y, Shelhamer E, Donahue J, et al. Caffe: Convolutional architecture for fast feature embedding[C]//Proceedings of the 22nd ACM international conference on Multimedia. ACM, 2014: 675-678.