Invasive Cancer Detection Utilizing Compressed Convolutional Neural Network and Transfer Learning

Bin Kong[1], Shanhui Sun[2](✉), Xin Wang[2], Qi Song[2], and Shaoting Zhang[1](✉)

[1] Department of Computer Science, UNC Charlotte, Charlotte, NC, USA
szhang16@uncc.edu
[2] CuraCloud Corporation, Seattle, WA, USA
shanhuis@curacloudcorp.com

Abstract. Identification of invasive cancer in Whole Slide Images (WSIs) is crucial for tumor staging as well as treatment planning. However, the precise manual delineation of tumor regions is challenging, tedious and time-consuming. Thus, automatic invasive cancer detection in WSIs is of significant importance. Recently, Convolutional Neural Network (CNN) based approaches advanced invasive cancer detection. However, computation burdens of these approaches become barriers in clinical applications. In this work, we propose to detect invasive cancer employing a lightweight network in a fully convolution fashion without model ensembles. In order to improve the small network's detection accuracy, we utilized the "soft labels" of a large capacity network to supervise its training process. Additionally, we adopt a teacher guided loss to help the small network better learn from the intermediate layers of the high capacity network. With this suite of approaches, our network is extremely efficient as well as accurate. The proposed method is validated on two large scale WSI datasets. Our approach is performed in an average time of 0.6 and 3.6 min per WSI with a single GPU on our gastric cancer dataset and CAMELYON16, respectively, about 5 times faster than Google Inception V3. We achieved an average FROC of 81.1% and 85.6% respectively, which are on par with Google Inception V3. The proposed method requires less high performance computing resources than state-of-the-art methods, which makes the invasive cancer diagnosis more applicable in the clinical usage.

1 Introduction

Invasive cancer is one of the leading worldwide health problems and the second killer in the United States [13]. Early diagnosis of invasive cancers with timely treatment can significantly reduce the mortality rate. Traditionally, the cancer regions of Whole Slide Images (WSIs) are delineated by experienced pathologists for histological analysis. However, precise delineation of the tumor regions and identification of the nuclei for pathologists are time-consuming and error-prone. Thus, Computer-aided diagnosis (CAD) methods are required to assist

© Springer Nature Switzerland AG 2018
A. F. Frangi et al. (Eds.): MICCAI 2018, LNCS 11071, pp. 156–164, 2018.
https://doi.org/10.1007/978-3-030-00934-2_18

pathologists' diagnostic tasks. However, WSI has a large image resolution up to $200,000 \times 100,000$ pixels and traditional methods are usually limited to only small regions of the WSIs considering the computational burden.

Recently, with deep learning becoming the methodology of choice [1,5], many work focus on applying these powerful techniques on directly analyzing WSIs. In the context of WSI cancer detection, the common practice is to extract patches of fixed size (e.g., 256×256) in a sliding window fashion and feed them to a neural network for prediction. In [16], GoogleNet [14] is utilized as the detector. Kong et al. [6] extended this approach considering neighborhood context information decoded using a recurrent neural network. Using model ensembles (i.e., several Inception V3 models [15]), Liu et al. [8] brings the detection result to 88.5% in terms of average FROC on CAMELYON16 [1] dataset. The above deep learning based methods face the common problem: the computation is expensive which becomes barriers to clinical deployment.

One of the most challenging problems for WSI image analytics is handling large scale images (e.g., 2 GB). High performance computing resources such as cloud computing and HPC machines mitigate the computational challenges. However, they either have a data traffic problem or are not always available due to high cost. Thus, high accuracy and efficiency become essential for deploying WSI invasive cancer diagnosis software into the clinical applications. In the previous work, there is few focusing on both accuracy and computation efficiency in computer-aided WSI cancer diagnosis. Resource efficiency in neural network study such as network compression is an active research topic. Product quantization [17], hashing, Huffman coding, and low bit networks [10] have been studied but they sacrifice accuracy.

How to make the invasive cancer detection system as efficient as possible while maintaining the accuracy? We answer this question with a suite of training and inference techniques: (1) For efficiency we design a small capacity network based on depthwise separable convolution [1]; (2) To improve accuracy, we refine the small capacity network learning from a large capacity network on the same training dataset. We enforced the logits layer of the small capacity network has a close response as logits layer of the large capacity network. A similar approach was investigated in work [2]. In addition to that, we use an additional teacher guided loss to help the small network better learn from the intermediate layers of the high capacity network; (3) To further speed up the computation in the inference stage, we avoid the procedure of frequently extracting small patches in a sliding window fashion but instead, we convert the model into a fully convolution network (a network does not have multilayer perceptron layers). As a result, our method is 5 times faster compared to one of the popular and state-of-the-art benchmark solutions without sacrificing accuracy.

In summary, the major contributions of this work are as follows: (1) We designed a multi-stage deep learning based approach to locate invasive cancer regions with discriminative local cues; (2) Instead of relying on histological level annotation such as nuclei centers and cell masks, the proposed method auto-

[1] https://camelyon16.grand-challenge.org/.

matically learn to identify these regions based on the regional annotation, which are much easier to obtain. Thus, our approach is extremely scalable. (3) We designed a novel method to shrink the large capacity network while maintaining its accuracy. (4) Our method is extensively validated on two large scale WSI dataset: gastric cancer dataset and CAMELYON16. The results demonstrate its superior performance in both its efficiency and accuracy.

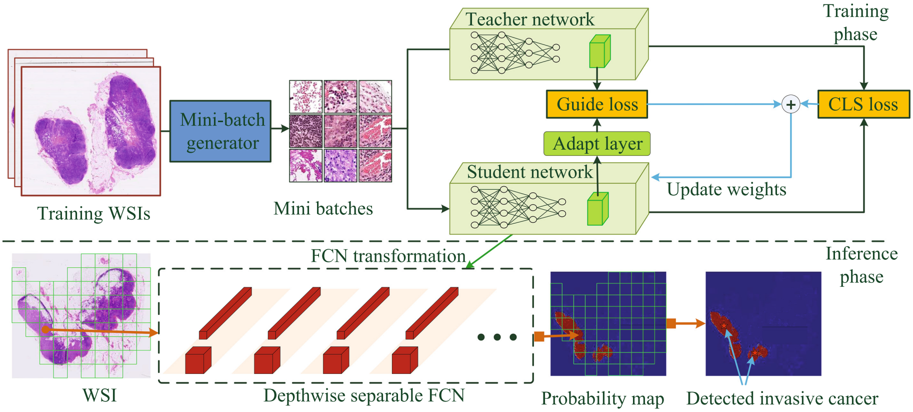


Fig. 1. Overview of the proposed framework. The above part indicates the training phase and below part indicates inference phase. Note that we only illustrate proposed transfer learning method in the training phase.

2 Methodology

2.1 Overview

Figure 1 shows an overview of our proposed method. The method is derived from detection by performing patch classification (normal patches vs cancer patches) via a sliding window. However, it is different from the traditional method of detection by performing classification. The base network is a small capacity network proposed for solving patch classification problem with a faster inference speed than a large capacity network.

This small network is trained on the training patches. The small capacity network has weak learning capability due to small number of learnable weights and may cause under-fitting and lower inference accuracy than the large capacity network in the inference stage. To solve this problem, we enforce the small capacity network learning the "useful knowledge" from the high capacity network in order to improve inference accuracy. Thus, we first train a high capacity network on the same training set. Then, we distill small capacity network's weights in a fine-tuning stage discussed in Sect. 2.3. In the inference stage, we convert multilayer perceptron layers (fully connected layers) of the network into fully convolution layers (fcn layers). This change allows the network using arbitrary sized tiles so that we can use large tiles resulting in faster speed. The output probability map is post-processed and detection results are produced from it using a method similar to [8].

The training objective function can be denoted as follows:

$$L = \frac{1}{|\mathbb{S}|} \sum_{x \in \mathbb{S}} (L_{cls}(x) + \lambda L_{guide}(x)) + \gamma L_{reg} \tag{1}$$

where $\mathbb{S}$ is the training patches. L_{cls} denotes the classification loss, comprising of the softmax loss using the hard ground truth label of the training patch and the regression loss using the soft probability label from the large capacity network. We will discuss it in detail in Sect. 2.3. L_{guide} is the teacher guided loss, which will be elaborated in Sect. 2.3. L_{reg} denotes the regularization penalty term which punishes large weights. Finally, λ and γ are balancing hyper-parameters to control the weights of difference losses, which are cross-validated in our experiments.

2.2 Small Capacity Network

To reduce the model's capacity, we utilized depthwise separable convolution in our small capacity network architecture. Depthwise separable convolution (depthwise convolution + pointwise convolution) is proposed in [1] and replaces convolution layers. Each kernel in a depthwise convolution layer performs convolution operation on only a single channel of input feature maps. To incorporate cross-channel information and change the number of output feature maps, pointwise convolution (i.e., 1×1 convolution) is applied after depthwise convolution. The depthwise separable convolution in [3] obtains a large factor of reduction in terms of computation comparing to corresponding convolution operations.

2.3 Transfer Learning from Large Capacity Network

We utilized a large capacity network (deep and wide network with more weights) to "teach" the small capacity network and adapt the model moving towards large network's manifold resulting logits of two networks being closer. We use the knowledge of both the output (probability) and intermediate layers (feature) in the large capacity network to teach the small capacity network.

Transfer Learning from the Probability: The network distilling technique proposed in [2] serves this transfer learning task. The softmax layer transforms the logit z_i for each class into the corresponding probability p_i:

$$p_i = \frac{\exp(z_i/T)}{\sum_{j \in \{0,1\}} \exp(z_j/T)} \tag{2}$$

where $i = 0$ and $i = 1$ represent negative and positive labels, respectively. T is the temperature which controls the softness of the probability distribution over the label. A higher temperature $T > 1$ produces soft probabilities distribution over classes, which helps the transfer learning. We used soft regression loss ($L_{soft} = ||p_i - \hat{p}_i||^2$, where p_i and $\hat{p}_i$ are the probabilities produced by the small and large capacity networks, respectively) to enforce small capacity network's outputs to match the large capacity network's outputs. We pre-trained the large capacity network using $T = 2$. In transfer learning, large capacity network's weights are fixed, and $T = 2$ is used in both small and large networks. In prediction, $T = 1$ is used.

We additionally use the hard ground truth label of the training patch to supervise the training. Then, the total classification loss is as follows:

$$L_{cls} = L_{hard} + \beta L_{soft} \tag{3}$$

where L_{hard} denotes the softmax (hard) loss. Hyper-paramter β controls the weights of hard and soft losses, which is cross-validated in our experiments.

Feature Adaptation from the Intermediate Layers: Romero et al. [11] demonstrated that the features learned in the intermediate layers of large capacity networks can be efficiently used to guide the student network to learn effective representations and improve the accuracy of the student network. Inspired by this idea, we apply the L2 distance between feature of the teacher network F_{tea} and the student network F_{stu}, which we name as teacher guided loss:

$$L_{guide} = ||F_{tea} - F_{stu}||^2 \tag{4}$$

While applying teacher guided loss, it is required that shape of the feature map dimension from teacher network should be the same as the student network. However, these two features are from different networks and the shape can be different. Thus, we use an adaptation layer (we use a fully connected layer) to map the feature from the student network to the same shape of the teacher network.

2.4 Efficient Inference

In most of popular WSI detection solutions such as [8,16], fixed-size patch based classification is performed in a sliding window fashion. The number of forward computation is linear to the number of evaluated patches. The memory cannot hold all patches so that frequent I/O operations have to be performed. This is the major source of the computational bottleneck. Inspired by [9], we replace all the fully connected layers in the small capacity network using equivalent convolutional layers. After the transformation, the FCN can take a significantly larger image if the memory allows. Let $size_p$ be the input image size used in a classification network before FCN transformation. After FCN transformation, the output of the network is a 2D probability map. The resolution of the probability map is scaled due to strided convolution and pooling operations. Let d be the scale factor. We assume that n layers (either convolution or pooling) have stride values >1 (i.e. stride = 2 in our implementation). Thus the scale factor $d = 2^n$. A pixel location $\mathbf{x}_o$ in the probability map corresponds to the center $\mathbf{x}_i$ of a patch with size $size_p$ in the input image. Centers displace d pixels from each other. $\mathbf{x}_i$ is computed as $\boldsymbol{x}_i = d \cdot \boldsymbol{x}_o + \lfloor (size_p - 1)/2 \rfloor$.

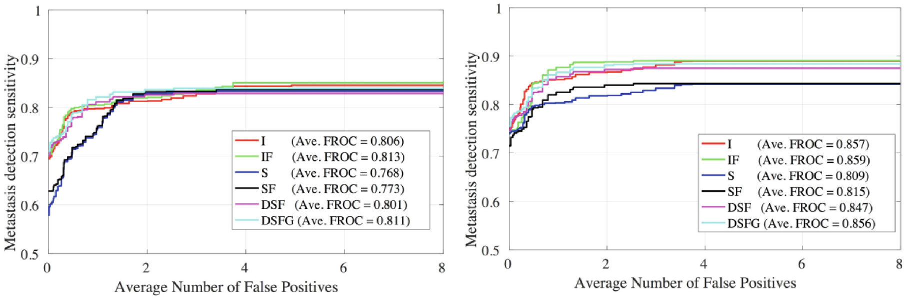
3 Experiment and Discussion

Datasets: Our experiments are conducted on gastric cancer dataset, acquired from our collaborative hospital. The invasive cancer regions were carefully delineated by the experts. It includes 204 training WSI (117 normal and 87 tumor) and 68 testing WSIs (29 tumor and 39 normal) with average testing image size 107595 × 161490. We additionally validated our approach on the CAMELYON16 dataset. It includes 270 training WSIs (160 normal and 110 tumor images), and 129 testing WSIs (80 normal and 49 tumor images) with average testing image size 64548 × 43633.

Experimental Setting and Implementations: We used Inception V3 as the teacher network (large capacity network) in transfer learning. To train the teacher network, we re-implemented the method in [8]. The patch size for the teacher network is 299 × 299. The patch size for the student network is 224 × 224. In the transfer learning, we randomly generated mini batches of patch size 299 × 299 for the teacher network and crop a 224 × 224 patch from it for the student network. We augment the training samples using random rotation, flipping and color jittering. We developed our approach using deep learning toolbox Caffe [4]. The inference part is implemented in C++ and validated on a standard workstation with a Nvidia Tesla M40 (12 GB GPU memory). To hide the I/O latency, we prefetch image patches into the memory in one thread and the network inference is implemented in another two threads. Note that this data prefetch scheme is applied to all investigated approaches. Besides this, we don't have other implementation optimization. In addition, all the experiments were conducted on the highest magnification (40×). For the student network, we use a normal convolutional layer followed by 13 depthwise separable convolutional (3 × 3 depthwise convolutional layer followed by 1 × 1 convolutional layer), 1 average pooling (7 × 7) and 1 fully connected layers. The number of convolution filters for the first to the last convolutional layers (including depthwise separable convolutional layers) are 32, 64, 128, 128, 256, 256, 512, 512, 512, 512, 512, 512, 960 and 960, respectively. We use average FROC (Ave. FROC, $[0, 1]$) [12] to evaluate detection performance. It is an average sensitivity at 6 false positive rates: 1/4, 1/2, 1, 2, 4, and 8 per WSI.

Table 1. Comparison for different detection approaches in terms of computation time and Ave. FROC. They are Inception V3, Inception V3 with FCN (IF), student network (S), student network with FCN (SF), distilled student network with FCN (DSF), and distilled student network with both FCN and teacher guided loss (DSFG, the final proposed network).

Methods		I	IF	S	SF	DSF	DSFG
Gastric cancer	Time (mins.)	3.8	2.3	1.5	0.6	0.6	0.6
	Ave. FROC	0.806	0.813	0.768	0.773	0.801	0.811
CAMELYON16	Time (mins.)	17.0	9.1	7.8	3.6	3.6	3.6
	Ave. FROC	0.857	0.859	0.809	0.815	0.847	0.856



(a) (b)

Fig. 2. Experimental results of the methods on the (a) gastric cancer and (b) CAMELYON16 datasets

Results and Analysis: We compared Inception V3 network (method I) using explicitly sliding window fashion proposed in [8], Inception V3 with our fully convolution (method IF) implementation, student network using explicitly sliding window (method S), student network with fully convolution (method SF), distilled student network with fully convolution (method DSF), and our final proposed approach: distilled student network with both FCN and teacher guided loss (DSFG). The stride of the sliding window is 128. The explicitly sliding window based method is the most widely used method and it achieved the state-of-the-art results [8,16]. Note that the original Inception V3 in [8] utilized 8 ensembled models. However, for a fair comparison here, we only used one single model. Due to GPU memory limitation, for the FCN based methods (IF, SF, DSF, and DSFG), we partition the WSI into several blocks with overlaps and stitch the probability maps to a single one accordingly after the inferences. In the method IF, we used a block 1451 × 1451 with an overlap of 267 pixels. In methods SF, DSF, DSFG, we used a block 1792 × 1792 with an overlap of 192 pixels.

Table 1 and Fig. 2 illustrate comparisons of these methods in terms of computation time and Ave. FROC. Fully convolution based detection significantly speeds up the inference compared to the corresponding sliding window approach. The method IF is 1.7 and 1.9 times faster than the method I for gastric cancer and CAMELYON16 datasets, respectively. The method SF is 2.5 and 2.2 times faster than the method S for gastric cancer and CAMELYON16 datasets, respectively. Note that small capacity model (SF) is about 2.5 and 2.2 times faster than the large capacity model (IF) for gastric cancer and CAMELYON16 datasets, respectively. In addition, we observed that the small capacity model reduced Ave. FROC of about 4% and 5% for gastric cancer and CAMELYON16 datasets, respectively. However, once the small network gained knowledge from transfer learning, the detection accuracy of it became close to the large model. For CAMELYON16 dataset, we observed that single Inception V3 model cause Ave. FROC decreasing to 85.7% from 88.5% reported in [8]. This drop is expected because ensembled models reduced model variance and overfitting. However, this result has been state-of-the-art accuracy among single model based methods. Lin et al. developed an efficient inference algorithm and in their study [7], they reported 15 min per WSI on CAMELYON16. While we achieved a much faster computation time, the validation is performed in different hardware and software environments.

These experiments demonstrate that we could keep the same detection accuracy compared to the method I and improve the efficiency significantly (5 times faster than the method I) via model "compression" and transfer learning. Our proposed model is more memory efficient and costs only 12 MB memory in contrast to the 84 MB required in the method I.

4 Conclusion

State-of-the-art deep CNN based invasive cancer detection methods have pushed the accuracy boundary closer to clinical application, however, the computation and memory burdens are barriers in real clinical setups. We proposed a new framework to keep high detection accuracy with efficient computation and memory usage. Particularly, we improved detection accuracy of a utilized small capacity network using a large capacity network pre-trained on the same data set, who taught the small network having similar prediction power. In addition, the proposed method requires less high performance computing resources and runs much faster than state-of-the-art methods. Thus, we expect that our work will become more applicable in the clinical usage.

Acknowledgements. This work is partially supported by the National Science Foundation under grant IIP-1439695, ABI-1661280 and CNS-1629913.

References

1. Chollet, F., et al.: Xception: deep learning with depthwise separable convolutions. In: CVPR, pp. 1251–1258 (2017)
2. Hinton, G., et al.: Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531 (2015)
3. Howard, A., et al.: Mobilenets: efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861 (2017)
4. Jia, Y., et al.: Caffe: convolutional architecture for fast feature embedding. In: ACMMM, pp. 675–678. ACM (2014)
5. Kong, B., Zhan, Y., Shin, M., Denny, T., Zhang, S.: Recognizing end-diastole and end-systole frames via deep temporal regression network. In: Ourselin, S., Joskowicz, L., Sabuncu, M.R., Unal, G., Wells, W. (eds.) MICCAI 2016. LNCS, vol. 9902, pp. 264–272. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46726-9_31
6. Kong, B., Wang, X., Li, Z., Song, Q., Zhang, S.: Cancer metastasis detection via spatially structured deep network. In: Niethammer, M., et al. (eds.) IPMI 2017. LNCS, vol. 10265, pp. 236–248. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-59050-9_19
7. Lin, H., et al.: Scannet: a fast and dense scanning framework for metastatic breast cancer detection from whole-slide images. arXiv preprint arXiv:1707.09597 (2017)
8. Liu, Y., et al.: Detecting cancer metastases on gigapixel pathology images. arXiv preprint arXiv:1703.02442 (2017)
9. Long, J., et al.: Fully convolutional networks for semantic segmentation. In: CVPR, pp. 3431–3440 (2015)
10. Rastegari, M., Ordonez, V., Redmon, J., Farhadi, A.: XNOR-net: imagenet classification using binary convolutional neural networks. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9908, pp. 525–542. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46493-0_32
11. Romero, A., et al.: Fitnets: hints for thin deep nets. arXiv preprint arXiv:1412.6550 (2014)
12. Shiraishi, J.: Computer-aided diagnostic scheme for the detection of lung nodules on chest radiographs: localized search method based on anatomical classification. Med. Phys. **33**(7), 2642–2653 (2006)
13. Siegel, R.L., et al.: Cancer statistics, 2017. CA Cancer J. Clin. **67**(1), 7–30 (2017). https://doi.org/10.3322/caac.21387
14. Szegedy, C., et al.: Going deeper with convolutions. In: CVPR, pp. 1–9 (2015)
15. Szegedy, C., et al.: Rethinking the inception architecture for computer vision. In: CVPR, pp. 2818–2826 (2016)
16. Wang, D., et al.: Deep learning for identifying metastatic breast cancer. arXiv preprint arXiv:1606.05718 (2016)
17. Wu, J., et al.: Quantized convolutional neural networks for mobile devices. In: CVPR, pp. 4820–4828 (2016)