Patch Efficient Convolutional Network for Multi-Organ Nuclei Segmentation and Classification

Hasib Zunair and A. Ben Hamza

Concordia Institute for Information Systems Engineering
Concordia University, Montreal, QC, Canada

Abstract. In this work, we introduce an end-to-end deep learning framework for automatic nuclei segmentation and classification from H&amp;E stained whole slide images (WSI) of multiple organs (breast, kidney, lung and prostate). The proposed approach, called PatchEUNet, leverages a fully convolutional neural network of the U-Net family by replacing the encoder of the U-Net model with an EfficientNet architecture with weights initialized from ImageNet. Since there is a large scale variance in the whole slide images of the MoNuSAC 2020 Challenge, we propose to use a patchwise training scheme to mitigate the problems of multiple scales and limited training data. For the class imbalance problem, we design an objective function defined as a weighted sum of a focal loss and Jaccard distance, resulting in significantly improved performance. During inference, we apply the median filter on the predicted masks in an effort to refine the segmented outputs. All codes are made available at URL.[1]

Keywords: Fully convolutional networks; multi-class segmentation and classification; efficient networks.

1 Problem Statement

If the rumour is tumour, the issue is tissue. In this paper, we tackle the problem of segmenting and classifying the type of cells present in a given H&E stained whole slide image (WSI) of multiple organs [5]. The cell types include epithelial cells, lymphocytes, macrophages and neutrophils. We pose this problem as multi-class segmentation defined as follows: given an input image, we determine for each pixel its semantic class $c \in \mathcal{C}$ in a single step with some probability, where $\mathcal{C} = \{1, 2, 3, 4, 0\}$ is a set of 5 classes: epithelial, lymphocytes, neutrophils, macrophages and background, respectively.

2 Dataset

The provided training data consist of 209 whole slide images of multiple organs (breast, kidney, lung and prostate) with cell-boundary and cell-type annota-

[1] Supplementary Material: https://github.com/hasibzunair/MoNuSAC-ISBI-2020

tions for epithelial cells, lymphocytes, macrophages and neutrophils. In total, the training data contain 31,411 hand-annotated nuclei instances, including 14,539 epithelial cells, 15,654 lymphocytes, 587 macrophages and 631 neutrophils. More details about the dataset can be found at [5].

For these 209 training images, we compute the mean and standard deviation of their height and width, yielding 563.9 ± 370.8 and 628.3 ± 408.9, respectively. The height and width of these images have maximum/minimum values of 1956/81 and 2162/74, respectively. Hence, the training dataset consists of images with varying scales. This issue becomes more challenging due to the already existing class imbalance problem. In order to address these issues, we employ a patchwise training scheme. We first extract overlapping patches of 96×96 using an adaptive sliding window technique with a step size of 16 for both images and the ground truths. For any patches resulting in a smaller size, we use reflection padding to resize and retain the original aspect ratio. Masks with a mean value of less than 0.099 were discarded from the training set as well as the corresponding image, as they mostly consist of background. We find this threshold by searching over different values and by visual inspection on a few WSIs. Then, we acquire $256,207$ patches of images and their corresponding masks over which we apply a 80:20 training/validation split. For training, we use $204,965$ images and their masks. For hyperparameter tuning, we use the remaining $51,242$ image and mask pairs.

3 Model

Our proposed method is based on the U-Net architecture [3], which consists of a contracting path that captures context and a symmetrically expanding path that enables precise localization. The contracting path follows the typical architecture of a convolutional network with alternating convolution and pooling operations and progressively downsamples feature maps, increasing the number of feature maps per layer at the same time. Every step in the expansive path consists of an upsampling of the feature map, followed by a convolution. Hence, the expansive branch increases the resolution of the output. In order to localize the upsampled features, the expansive path combines them with high-resolution features from the contracting path via skip-connections [3]. The output of the model is a pixel-by-pixel mask that shows the class of each pixel.

In this work, we scale the encoder of the U-Net architecture and initialize it with weights from the ImageNet dataset. More specifically, we use the EfficientNet-B3 architecture [4], which results from a compound scaling method applied on the baseline network EfficientNet-B0 that uniformly scales all three dimensions with a fixed ratio. The architecture consists of seven blocks which multiple operations the hierarchically downsample the input. In the decoder stage, we use five blocks to order increase the grid size of the feature maps. Essentially, the reverse of the downsampling path is carried out. Each decoder block consist of an upsampling layer and convolutional layer with batch normalisation and ReLu activation. We use 256, 128, 64, 32 and 16 filters for the convolutional layers in the decoder blocks. Each decoder block is combined to output of the EfficientNet-B3 blocks numbered 2, 3, 4 and 6 via skip-connections.

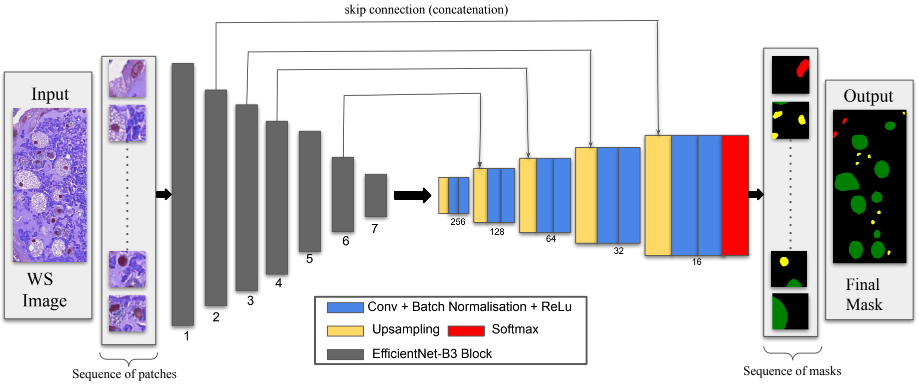


Fig. 1. An illustration of the proposed PatchEUNet architecture. The whole slide image is divided into a sequence of patches. The encoder shows the architecture blocks consisting on multiple operations. The convolutional filters used for each layer in the decoder block are numbered. The final decoder output feature maps are fed into a softmax classifier for pixel-wise classification. Finally, the patch level predicted masks are stitched together to output the final segmentation mask.

4 Implementation Details

4.1 Training

As the first step in training, we prepare masks as five channel images using one-hot encoding. For the evaluation metric, we use the F-score, which is defined as the harmonic mean of precision and recall. An F-score reaches its best value at 1 and worst score at 0. A performance measure commonly used for evaluating segmentation masks is the Jaccard index, also known as the intersection-over-union (IoU) score. Given a vector of ground truth labels $\mathbf{y}$ and a vector of predicted labels $\hat{\mathbf{y}}$, we define the multi-class Jaccard index as follows

$$\mathcal{J}(\mathbf{y}, \hat{\mathbf{y}}) = \sum_{c=1}^{4} w_c \frac{|\{\mathbf{y} = c\} \cap \{\hat{\mathbf{y}} = c\}|}{|\{\mathbf{y} = c\} \cup \{\hat{\mathbf{y}} = c\}|}, \tag{1}$$

where w_c denotes the weight for class c, and is used to help prevent difficulties with class imbalance. We set class weights $w_c = \{0.5, 0.5, 2, 2\}$ for $c \in \{1, 2, 3, 4\}$. The multi-class Jaccard index is computed based on four target classes, i.e. we exclude the background class.

Since the image segmentation task can be considered as a pixel classification problem, we use the focal loss function defined as

$$\mathcal{F}(\mathbf{y}, \hat{\mathbf{y}}) = -\frac{1}{N} \sum_{i=1}^{N} \sum_{c=1}^{4} \alpha (1 - \hat{y}_{ic})^{\gamma} y_{ic} \log(\hat{y}_{ic}), \tag{2}$$

where N is the number of training samples, $\hat{y}_{ic}$ is the probability that the network associates the i-th sample with class c, α is a weighting factor and γ is a tuneable focussing parameter. We set $\alpha = 0.25$ and $\gamma = 2$ as in [2]. The focal loss function assists the model in learning from hard examples, while down-weighting the easy ones.

We convert the Jaccard similarity index into a dissimilarity measure in order to combine it with the focal loss, and then optimize a weighted loss function given by

$$\mathcal{L}(\mathbf{y}, \hat{\mathbf{y}}) = \lambda \, \mathcal{F}(\mathbf{y}, \hat{\mathbf{y}}) + \mu \, (1 - \mathcal{J}(\mathbf{y}, \hat{\mathbf{y}})), \tag{3}$$

where λ and μ are regularization parameters. By minimizing $\mathcal{L}$, we simultaneously maximize the probabilities for right pixels to be predicted and maximize the intersection between masks and corresponding predictions. For the multi-class segmentation problem, we found that $\lambda = 1$ and $\mu = 0.5$ yield the best performance on the validation set.

For training and validating our network, we perform a 80:20 split on the $256,207$ images and mask pairs. We train our network using Adam solver [1] with learning rate 1e-4, decay 1e-4 and a batch size of 128. A lower batch size resulted in degraded performance, while increasing the batch size resulted in GPU memory error. This issue was observed over multiple runs. In all experiments, the training and inference are carried out using a single RTX 2080Ti GPU. Training is continued on all network layers until the intersection over union in the validation set stops improving, and then the best weights are retained.

4.2 Inference

During inference, we perform predictions by using a non-overlapping adaptive sliding window technique. From experimentation, we found that overlapping windows and averaging predictions do not improve validation results. We slide over the input image extracting patches of size 96×96 with a step size of 96. Similar to the patchwise training scheme, we pad any input patch less than 96×96, compute the prediction mask and then un-pad the mask to the original size in order to stitch all patches to construct the final high resolution predicted mask. As a post processing step, we apply the median filter to further refine the segmentation masks.

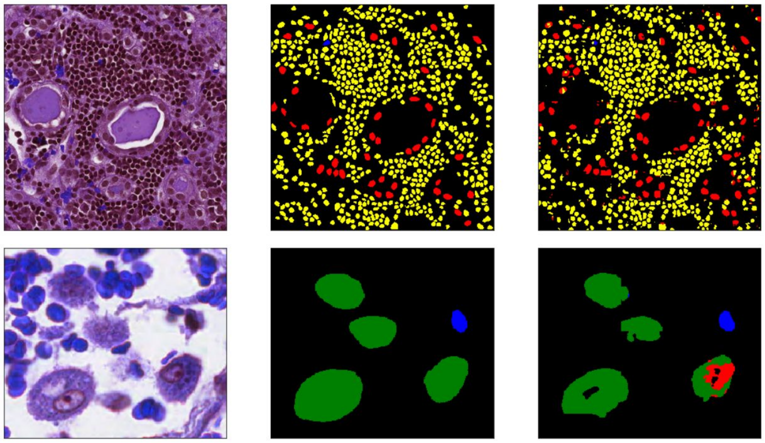
The last step during the inference is to post process predicted binary masks and touching borders in such a way that the binary mask is split into separate instances. To make this, we subtract touching borders from the corresponding mask to obtain seeds and use both masks and these newly generated seeds as an input to the watershed transform.

5 Results

In Table 1, we summarize our results on the validation set using the IoU and F-score metric. From this table, it is evident that the using a compound scaled decoder, EfficientNet-B3, significantly improves the performance. The proposed PatchEUNet yields improvements by optimizing the weighted multi-task loss function. It is important to note that for both U-Net and EfficientNet-B3 U-Net, we only use the Jaccard index. In Figure 2, we show some qualitative results that demonstrate the performance of the proposed approach.

Table 1. Performance metrics in % reported on the validation set consisting of 77,073 image mask pairs.

Configuration	**IoU**	**F-score**
U-Net	77	78
EfficientNet-B3 U-Net	82	83
PatchEUnet	**84**	**87**



a) Input whole slide image

b) Ground truth mask

c) Predicted mask

Fig. 2. Qualitative results for the proposed model: a) original whole slide image; b) corresponding ground truth annotation; and c) predicted mask by the model.

References

1. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
2. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: Proceedings of the IEEE international conference on computer vision. pp. 2980–2988 (2017)
3. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention. pp. 234–241. Springer (2015)
4. Tan, M., Le, Q.V.: Efficientnet: Rethinking model scaling for convolutional neural networks. arXiv preprint arXiv:1905.11946 (2019)
5. Verma, R., Kumar, N., Patil, A., Kurian, N., Rane, S., Sethi, A.: Multi-organ nuclei segmentation and classification challenge 2020 (02 2020). https://doi.org/10.13140/RG.2.2.12290.02244/1