CONVOLUTIONAL NEURAL NETWORKS AND RANDOM FORESTS FOR DETECTION AND CLASSIFICATION OF METASTASIS IN HISTOLOGICAL SLIDES

Klaus Strohmenger[1], Jonas Annuscheit[1]
Iris Klempert[2], Benjamin Voigt[1],
Christian Herta[1], Peter Hufnagl[1,2]

[1]Hochschule für Technik und Wirtschaft Berlin, Univeristy of Applied Sciences
[2]Charité – Universitätsmedizin Berlin

ABSTRACT

For many malignant (metastatic) tumors, e.g. for breast cancer, one way - and also the most likely way - of spreading, is the lymphatic system. For treatment decisions, it is essential to examine the lymph nodes surrounding the primary tumor with a microscope for metastatic infiltration, which is a time-consuming and error-prone process. A pN-stage designation summarizes the number of affected lymph nodes and the severity of the infiltration. Today, slides with tissue can be digitized at a quality which corresponds approximately to the current state of light microscopy. These digitized slides are called Whole Slide Images (WSIs). In this work, we present a proof of concept, which completely automates the process of analyzing and classifying these WSIs and further predicting the patient's pN-stage. To accomplish this, we evaluated Convolutional Neural Networks, Fully Convolutional Networks, Multi Layer Perceptrons and Random Forests on the CAMELYON 16 and 17 dataset and combined the most promising techniques.

***Index Terms*—** CAMELYON Challenge, Convolutional Neural Network, Random Forest, Whole Slide Images, Metastasis Detection

1. INTRODUCTION

Approximately 12% of all women get breast cancer or one of its precursors in their lives. Therapy response and overall prognosis are highly dependent on tumor size, lymph node infiltration and metastasis status (see [1, p.379ff]). The *pN* stage describes the status of lymph node infestation and is part of the TNM system (Tumor Nodes Metastasis) for the staging of malignant tumors, which in turn is the primary basis for the subsequent therapeutic decisions. *T* describes the primary tumor, *N* the lymph node infiltration and *M* the metastasis status (see [2, p.108f]).

The pN-stage (pathologic N-stage) is determined by examining the surrounding lymph nodes of the primary tumor. If a metastasis region is detected in one of the slides of theses lymph nodes, the cells contained in that region are counted and/or the size of the region is measured. Regions containing less than 200 cells or being smaller than $0.2mm$, are classified as *itcs* (isolated tumor cells). Regions containing more than 200 cells and sized in between $0.2mm$ and $2.0mm$, are classified as *micrometastasis*. Greater regions are called *macrometastasis*. The number of slides classified into each of these classes, then determines the patient's pN-stage. Table (1) shows a subset of pN-stages and provides a short description.

Table 1. The pN-stages used by the CAMELYON17 Challenge.

pN-stage	description
pN0	No metastasis found
pN0(i+)	Only itcs found.
pN1mi	Micro metastasis found, but no macro metastasis
pN1	Metastasis found in 1-3 lymph nodes. At least 1 is a macro metastasis
pN2	Metastasis found in more than 3 lymph nodes. At least 1 is a macro metastasis

The described process is performed by pathologists who need years of training to identify the abnormalities in the tissue. Nevertheless, it remains a time-consuming and error-prone process. The use of artificial intelligence in this field has great potential to save time, resources and to improve quality assurance.

2. METHODOLOGY

On the following pages, we present a proof of concept to predict the patient's pN-stage automatically. Several machine learning techniques, like Convolutional Neural Networks (CNNs), Random Forests [3] and Multi Layer Perceptrons (MLPs), have been evaluated. Finally, the best performing methods were chosen and combined together. Figure (1) shows an overview of our final concept. All tests were performed using the CAMELYON16 and CAMELYON17 dataset.

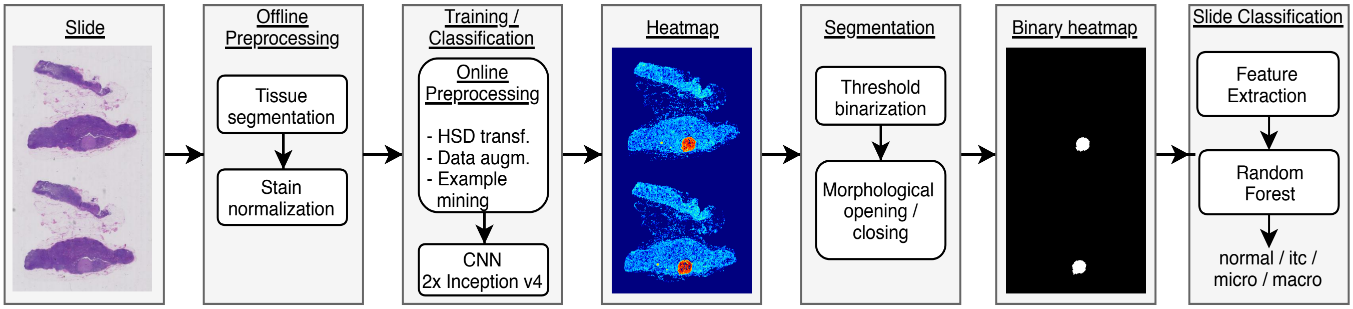

Fig. 1. Overview of the proposed framework. Patient pN-stage classification is left out here, because it is a simple rule based assignment once the WSI labels *normal*, *itc*, *micro* and *macro* are obtained.

2.1. Dataset

The CAMELYON Challenge provides two datasets: The CAMELYON16 and the CAMELYON17 dataset. The CAMELYON16 training set consists of 270 WSIs, annotated with the labels *normal* and *tumor*. For WSIs labeled tumor, masks are provided, localizing the metastasis regions. The CAMELYON16 test set consists of 130 WSIs, which are further annotated with the labels *normal*, *micro* and *macro*. The CAMELYON17 training set consists of 100 patients with five WSIs per patient, labeled with *normal*, *itc*, *micro* and *macro*. The CAMELYON17 test set also contains 100 patients with five WSIs per patient. The labels of the test set are unknown to all participants of the challenge.

2.2. Preprocessing

For many WSIs, the area containing tissue only makes up a small portion of the whole image. Therefore the first step was to detect and extract these regions. For this task we used the OTSU threshold algorithm [4]. In average we could reduce the size of the WSIs down to 16.3%, without losing areas containing tissue. We used the proposed algorithm [5] for stain normalization and creating look-up tables (LUTs) for regions of the WSI, storing information on how to normalize them later in the training and classifying process. We want to outline that this is a crucial task, as the WSI's colors from different facilities and different slide scanners can differ greatly. As shown in figure (1), all this was done before the training phase.

During training we applied stain normalization, using the LUTs created beforehand. Since the dataset is heavily unbalanced, containing much more WSIs labeled with *normal* and even WSIs labeled with *tumor* mostly contain normal tissue, we used hard example mining. Further we applied data augmentation by *rotating*, *mirroring* and *single pixel manipulation*.

2.3. Heatmap generation

To turn WSIs into heatmaps, we used two Inception-v4 models as CNNs [6]. Both receive patches from the WSIs, sized $312x312$. One network receives the highest resolution (x40) a pathologist typically can use and the other the second highest resolution (x20). At the end, both networks are combined with a fully connected layer. This technique was introduced in [7].

We have modified the Inception-v4 models and extended them by several layers. First we changed the first three convolutional layers and turned them into atrous layers [8] to get more overlapping pixels. Then we added the roll, stack and slice layers to the network, which were proposed in [9], in order to make CNNs partially rotation equivariant.

For the generated heatmaps one pixel presents a patch of 256x256 pixel of the maximum resolution of the original WSI. On average, our model produces six heatmaps per hour using two Geforce Titan X GPUs.

2.4. WSI classification

Overall we extracted 10 Features from the heatmaps. Some features could be extracted on the raw heatmap. These were *(1) highest probability*, *(2) sum of all probabilities* and *(3) avg probability*. For the remaining seven features the heatmap had to be segmented before. The segmentation process consisted of applying a global threshold t, applying morphological opening o times and morphological closing c times. We introduce a method to determine the quality of this postprocessing process before applying a classifier:

1. Subtract the pathologists masks from the binarized heatmap, whichs yields an error mask.

2. Sum the error pixels and divide by the number of pixels of the pathologists mask.

3. Sum over all processed heatmaps.

On the segmented heatmaps we extracted the features *(4) total area*, *(5) major axis of the largest connected region*, *(6)*

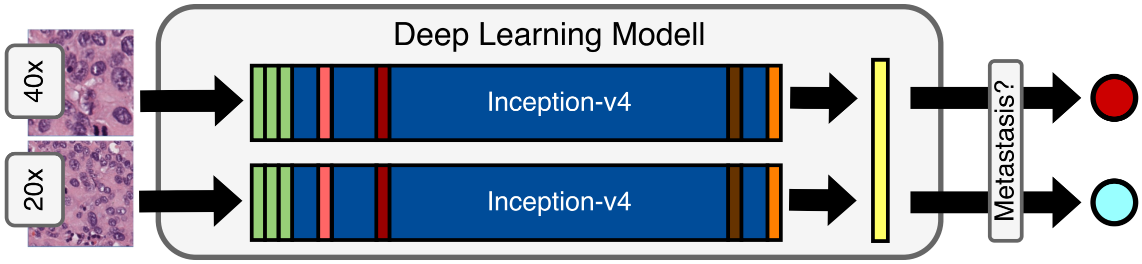

Fig. 2. Construction of the final trained CNN model. Three Atrous layers (green), a slice layer (red), a roll layer (dark red), a stack layer (brown) and a fully connected layer (orange) were used in one Inception-v4 model. Two models, each working with a different resolution and receptive field were used and combined with a fully convolutional layer.

minor axis of the largest connected region, *(7) area of the largest connected region*, *(8) major axis of the second largest connected region*, *(9) minor axis of the second largest connected region* and *(10) area of the second largest connected region*.

Since almost all other participating teams in the CAMELYON17 Challenge used a *Random Forest* for WSI classification, we decided to also use this algorithm, at least as a baseline for comparison with other methods. The next classification algorithm we evaluated for WSI classification were *MLPs*, which, just like the *Random Forests*, were trained with the 10 extracted features mentioned. Furthermore, several *CNNs* were evaluated with the raw heatmaps as input, instead of the manually extracted features. The dataset consisted of the CAMELYON17 training set and the CAMELYON16 test set, containing a total of 782 WSIs. Because of the small size of the dataset, especially only including 35 WSIs labeled with *itc*, we used 10-fold-cross-validation.

The last task, the pN-stage classification for each patient was a simple rule based assignment once the WSI's labels had been obtained.

3. RESULTS

We used the CAMELYON16 test dataset to evaluate the quality of the generated heatmpas. In order to compare our results, we used the two metrics, which were also used in the CAMELYON16 Challenge: The AUC for the ROC curve if a WSI contains metastasis or not and the AUC for the FROC curve, whether each pixel in a WSI is metastasis or not. For the ROC's AUC we achieved 0.983, which would have been the second best score out for 32 participating teams. For the FROC's AUC we measured 0.639, which would have been rank 6. We want to outline that we did not apply any postprocessing to the heatmaps before measuring the FROC's AUC, so there is still potential for increasing this value.

The CAMELYON17 Challenge then uses the quadratic weighted kappa score, where the classes are the pN-stages. For internal comparisons we also used the accuracy for the classification of the individual WSIs, since it is very likely to achieve a high kappa score by accident, when evaluating a high number of different postprocessing and/or classifier settings. In total we evaluated over 240 different parameter settings in the postprocessing of the heatmaps, each with over 25 parameter settings for the WSI classifier. Random Forests seem to work best with $t = 0.4$, $o = 0$ and $c = 0$. MLPs seem to prefer a higher threshold $t = 0.6$ to 0.7, combined with morphological operations $o = 2$ and $c = 2$. The best performing CNN for WSI classification was a heavily simplified AlexNet [10] without fully connected layers. The simplification consisted of reducing the number of kernels in each layer to prevent overfitting. The overall best classifier was a Random Forest with 1.500 trees and a maximum tree depth of 4, achieving a kappa score of 0.850. Table (2) shows the best results for each classification algorithm.

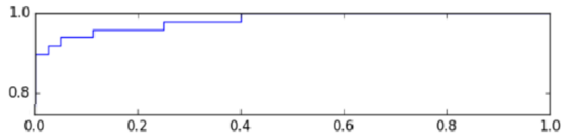

Fig. 3. The ROC curve shows the results for the CAMELYON16 test dataset using the evaluation script from the CAMELYON16 website. The ROC's AUC was 0.9829.

Table 2. Best results for each classification algorithm for pN-stage classification (kappa score, CAMELYON17 training set only) and WSI classification (accuracy, CAMELYON17 training set and CAMELYON16 test set) using 10-fold-cross-validation.

Classifier	Kappa score	Accuracy
Random Forest	0.850	0.871
MLP	0.822	0.870
CNN	0.752	0.801

The proposed formula in (2.4) for calculating a postprocessing error of the heatmaps turned out to be useful. Figure (4) shows the context of kappa score and the postprocessing error.

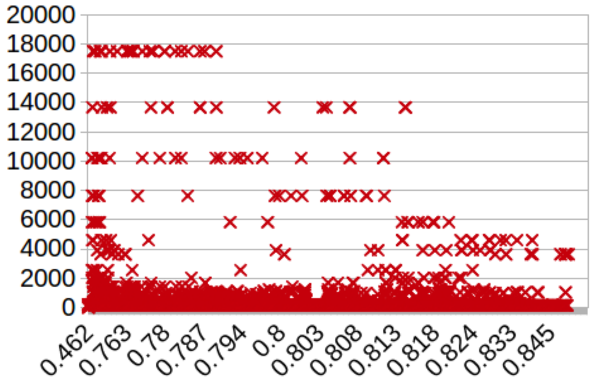

Fig. 4. Context between kappa score (x axis) and postprocessing error (y axis). Results of more than 6.000 different parameter settings are shown.

4. DISCUSSION

For random forests in the WSI classification, morphological operations turned out not to be useful. We estimate, that noise in the heatmaps has a relatively low impact on results, since the only feature influenced by it is *(4) total area* (after thresholding).

Our estimation that a CNN would not be able to compete with Random Forests for WSI classification was met. Nevertheless, a kappa score of 0.752 is beyond expectations and shows that this approach could become a viable option with more training data available.

5. CONCLUSION

In this article we presented a way to generate heatmaps for WSIs and how to use them to predict the WSIs labels (*normal*, *itc*, *micro* and *macro*) and the patient's *pN-stage*.

Improvement possibilities are seen in the postprocessing of the heatmaps. It has been found that by careful selection of the binarization threshold, morphological operations to connect nearby regions are superfluous. Instead of morphological operations, smarter methods should be tested. Possible solution approaches to segmentation could be the use of conditional random fields, fully convolutional networks or clustering algorithms.

6. ACKNOWLEDGEMENTS

We would like to thank Babak Ehteshami Bejnordi for his algorithm for stain normalization.

7. REFERENCES

[1] H. Genzwürker, J. Hinkelbein, J. Keil, G. Zimmer, and H. Ackermann, *AllEx - Alles fürs Examen: Das Kompendium für die 2. ÄP*, Thieme, 2014.

[2] Gerd Herold, *Innere Medizin - eine vorlesungsorientierte Darstellung ; unter Berücksichtigung des Gegenstandskataloges für die Ärztliche Prüfung ; mit ICD 10-Schlüssel im Text und Stichwortverzeichnis*, Selbstverl., Berlin, 2007.

[3] Leo Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, Oct 2001.

[4] Nobuyuki Otsu, "A Threshold Selection Method from Gray-level Histograms," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.

[5] Babak Ehteshami Bejnordi, Geert Litjens, Nadya Timofeeva, Irene Otte-Höller, André Homeyer, Nico Karssemeijer, and Jeroen AWM van der Laak, "Stain specific standardization of whole-slide histopathological images," *IEEE transactions on medical imaging*, vol. 35, no. 2, pp. 404–415, 2016.

[6] Christian Szegedy, Sergey Ioffe, and Vincent Vanhoucke, "Inception-v4, inception-resnet and the impact of residual connections on learning," *CoRR*, vol. abs/1602.07261, 2016.

[7] Yun Liu, Krishna Gadepalli, Mohammad Norouzi, George E Dahl, Timo Kohlberger, Aleksey Boyko, Subhashini Venugopalan, Aleksei Timofeev, Philip Q Nelson, Greg S Corrado, et al., "Detecting cancer metastases on gigapixel pathology images," *arXiv preprint arXiv:1703.02442*, 2017.

[8] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected crfs," *CoRR*, vol. abs/1412.7062, 2014.

[9] Sander Dieleman, Jeffrey De Fauw, and Koray Kavukcuoglu, "Exploiting cyclic symmetry in convolutional neural networks," *arXiv preprint arXiv:1602.02660*, 2016.

[10] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds., pp. 1097–1105. Curran Associates, Inc., 2012.