Parallel Computing: Technology Trends
I. Foster et al. (Eds.)

doi:10.3233/APC200022

First Experiences on Applying Deep Learning Techniques to Prostate Cancer Detection

Eduardo José GÓMEZ-HERNÁNDEZ [a,1] and José Manuel GARCÍA [a]
[a] *Computer Engineering Department, University of Murcia, Murcia, Spain*

Abstract. Nowadays, machine learning techniques based on deep neural networks are everywhere, from image classification and recognition or language translation to autonomous driving or stock market prediction. One of the most prominent fields of application is medicine, where AI techniques promote and promise the personalized medicine. In this work, we entered in this field to study the prostate cancer prediction from images digitalized from hematoxylin and eosin stained biopsies. We chose this illness since prostate cancer is a very common type of cancer and the second cause of death in men. We did this work in collaboration with Hospital Reina Sofia of Murcia. As newcomers, we faced a lot of problems to start with, and questioned ourselves about many issues. This paper shows our experiences in developing and training two convolutional neural networks from scratch, exposing the importance of both the preprocessing steps (cropping raw images to tiles, labeling, and filtering), and the postprocessing steps (i.e., to obtain results understandable for doctors). Therefore, the paper describes lessons learned in building CNN models for prostate cancer detection from biopsy slides.

Keywords. Deep learning, Prostate cancer detection, Neural networks

1. Introduction

Deep learning has become more and more ubiquitous in everybody's day life. Specifically, deep neural networks have been incorporated into numerous fields, such as image classification, language processing, economics, video games, and medicine. In some tasks, this new technology is being able to outperform human performance.

Progress in hardware technologies and cost reduction have caused new approaches in deep neural networks, outperforming older machine learning techniques. Nowadays it is possible to afford more and more complex problems, then it is important to have some practice and experience on it.

One of the most prominent fields of application of deep learning techniques is in medicine. Initially, applications of deep learning in medicine were limited to radiology images [1], but later (since end of 2016) it began to apply for other kinds of images [2–5].

Medical image analysis has started to implement deep learning for screening and localization of malignant zones. Additionally, other medical areas are working with these

[1] Corresponding Author: Computer Engineering Department, University of Murcia, 30.100 - Murcia (Spain); E-mail: eduardojose.gomez@um.es.

kind of techniques as well, like the analysis of the genetic information inside DNA and RNA series [6]. The common objective is not replace physicians with deep learning techniques, but supporting them to make better diagnoses.

Although our research group is focused on High-Performance Computing (HPC) and its applications, some years ago we got attracted by using our knowledge on HPC techniques to improve the precision and execution time of deep learning workloads from a real medical case. Then, we joined Prof. Enrique Poblet-Martínez and his research team from the Hospital Reina Sofía of Murcia to work in the field of "Detection of Prostate Cancer by biopsy slides". The final objective of our new research line is to create a model able to recognize tumorous zones in biopsy slides as a preliminary screening, hence allowing doctors to focus on the tumorous cases.

This paper presents our first experiences in this field, showing the way we took to learn about Deep Neural Networks (DNNs) tackling a real problem from the medicine field. We started by choosing the MXNet framework as our platform where our codes have been run. The first lesson we learned was the major role that the preprocessing steps play to observe a good behaviour of the CNN network . Next, we realized about modifying the hyper-parameters of the network to tune its behaviour and further improve its "accuary". Finally, we discovered the importance of the postprocessing steps to present the neural network's output in a format understandable by pathologists. In this first attempt, we achieved an AUC satistic metric of 82% in discriminating healthy from cancerous images using Inception V3.

The rest of the paper is organized as follows: Section II introduces the main concepts managed throughout this paper, related to deep learning frameworks, accelerators, and prostate cancer. Section III reports the machine and configuration used. The methodology is described in Section IV. Section V contains the obtained experimental results. Finally, Section VI exposes our conclusions and give some hint for future work.

2. Background

2.1. Machine Learning & Deep Learning

Theoretical and mathematical models of the artificial intelligence techniques were developed in the twentieth century. One of these models are ANNs (Artificial Neural Networks), a type of brain-inspired learning algorithm, built from small units called neurons. The most classical one, MLP (MultiLayer Perceptron) network, With enough layers, and enough perceptrons per layer, is able to represent any mathematical function [7]. However, when the amount of data grows, networks built exclusively from perceptrons can be very inefficient. Therefore, new types of neural networks should be made, being CNNs (Convolutional Neural Networks) the most known ones. The most distinguished layers in these networks are convolution and pooling, taking input data structured as channels of two dimensions.

Once the network is defined, with more or fewer layers, there are two different phases: inference, and training. In the inference phase, a set of inputs are presented to the network, and a set of outputs are given by the network, like any mathematical function. But training phase is more complicated, using an algorithm, it starts to teach the network to do something useful.

Before starting training, a initialization step is needed to set the different parameters of the network. This step might appear trivial or optional, but a bad starting point may make the network never be able to learn. Also, it is possible to bring this data from another neural network model, called Transfer Learning [8].

SGD (Stochastic Gradient Descendent) is the most known training algorithm for neural networks, but it is not the only one, there others like Adam [9] and DCASGD [10] among others. SGD is a variant of GD (Gradient Descendent) but used with batches. A batch is a group of input data of fixed size.

Then, the iteration process is as follow: First the Feed-Forward step, where data is presented to the network in batches, storing the result for later use. Then, the Back-Propagation step that compares all the results from the previous step with ground truth, and propagates backward on the network to calculate the gradient estimate. Finally, with the gradient estimation, all weights and biases are updated in the Update step.

There are some metrics to observe the precision of the neural network, being the most common accuracy, mse, macc, and cross-entropy. In classification, the most used is accuracy, giving the percentage of correctly predicted cases over the total. In classification, AUC statistic metric has started to be used in neural networks for medicine. AUC is the Area Under the Curve, to be specific, under the ROC curve. A ROC curve is a Receiver Operating Characteristic Curve [11], and it is commonly used to know how good is a binary classifier.

2.2. Frameworks & HPC

Machine learning techniques could be difficult to code and debug, therefore many frameworks have been developed to ease its use. Most of them are open source with software for most of the types of neural networks. The most known ones are Caffe, Caffe2, Tensorflow, Theano, PyTorch, Mxnet, and CNTK among others [12]. And there are frameworks like Keras, providing a more high-level experience, running on top of some of the aforementioned frameworks.

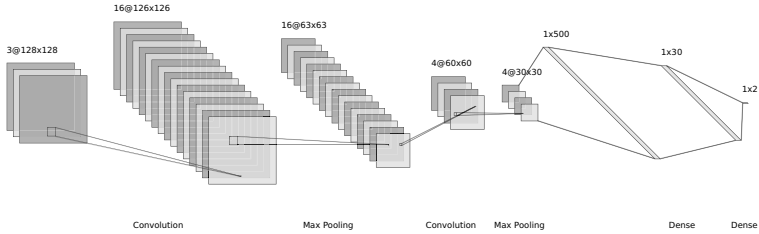
Specifically, the training phase is very time-consuming, since it is evaluating an optimization problem with hundreds, thousands, or even millions of parameters. Therefore, the reduction of the training phase execution time is a desirable feature for all frameworks. Thanks to this shorter training time, scientists using theses frameworks can explore a wide solution space, and even develop more complex networks.

The rise of High-Performance Computing (HPC) applied not only to grand challenge problems but also to common problems has revolutionized the machine learning field. All major vendors offer products which can be used for deep learning, as GPUs from Nvidia and AMD or scalable Xeons from Intel. Also, proprietary designs have emerged using ASICs, FPGAs or systolic arrays. Maybe the most well known is the development of the TPUs from Google [13].

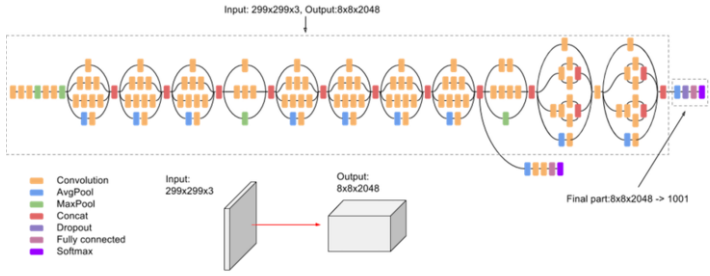
2.3. Prostate Cancer

Prostate cancer is the most common cancer and the second leading cause of death in men [14]. Nowadays, pathologists have a large number of slides to diagnose, making diagnosis very long. Reduce this diagnosis time would help to focus on the needed cases.

Prostate biopsies are hematoxylin and eosin stained (H&E) and normally stored inside a crystal. These biopsies need to be transformed to digital images to be used by



(a)



(b)

Figure 1. a: **Custom Neural Network** inspired on LeNet. **b**: **Inception V3** https://cloud.google.com/tpu/docs/inception-v3-advanced.

deep learning techniques. To do that, there are systems able to scan biopsies into a high-resolution image, called WSI (whole slide images). These WSI allow the application of image analysis techniques to prostate biopsies.

Using these WSIs, there are some approaches developing deep learning models to detect prostate cancer in biopsy slides [15–17]. Some of them try to find the tumorous zones and classify them using the Gleason's pattern.

3. Our Experience

3.1. Settings

In this study, we have used the MXNet 1.3.0 framework running with Cuda 9.2, and cuDNN 7.4.1. Statistical data was obtained with scikit-learn 0.20.2. Our compute machine is running CentOS Linux 7.5.1804 with Linux 3.10.0-862.14.4, powered by two Intel(R) Xeon(R) CPU E5-2603 v3 @ 1.60GHz with 64 GiB RAM memory, and a Geforce GTX 1080 8GB GDDR5X. For storage, we have a 500GB Samsung SSD 850. Finally, the scanner used to digitalize biopsies was iScan Coreo Ventana, able to produce BIF, TIFF and JPEG2000 image formats, from 1x to 40x magnification.

Two neural networks architectures have been used. The first one is a basic CNN (Figure 1) based on LeNet [18] and previously used in a medical environment [2, 19]. The second one is Inception v3, a very common network used for image classification (Figure 1). To clarify these figures, next we detail the different layers from the LeNet-based network. At the beginning is the input image (3 channels of 128x128 pixels); then

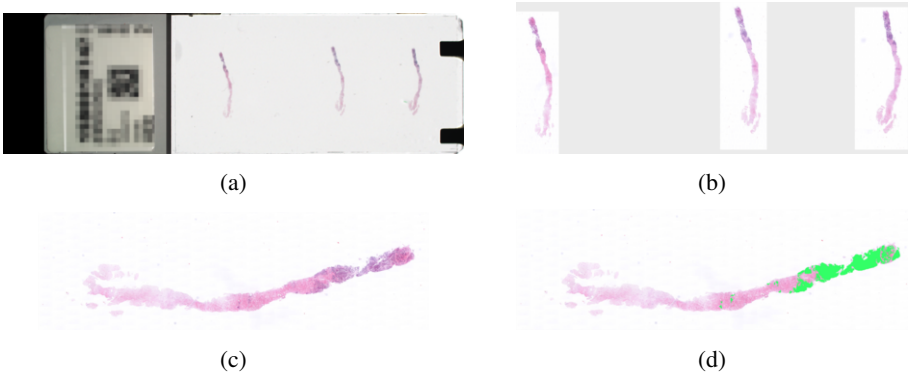


Figure 2. An example of a biopsy. a: a crystal with 3 slabs of a biopsy. **b**: scanned biopsy with all slabs. **c**: a slab extracted from the scanned image. **d**: a labeled slab of a biopsy, green zones denote tumors.

a convolution of 16 filters with a kernel size of 3x3; its size is reduced by a 2x2 max pooling layer; then, another convolution is applied, but with a kernel size of 5x5 and 4 filters, followed again with another 2x2 max pooling; in the end, we have an MLP with sizes 500, 30, and 2, each one with sigmoid activation; and a SoftMax layer at the end.

3.2. Preprocessing Database

As mentioned before, WSI images have a high resolution scanned image. Our selected framework, MXNet, cannot use this multiple format. Then, we chose the TIFF format to convert it later to JPG. These TIFFs have multiple layers, the first one is an image of the biopsy (Figure 2), the next one is a thumbnail, and the consecutive ones are 20x, 10x, 5x, 2.5x, 1.25x, 0.625x, 0.3125x, 0.15625x, 0.078125x magnifications.

The start point was to set a magnification value for the images. Pathologists usually select 20x magnification to find tumors, therefore we extract this specific image from the multilayer TIFF image (Figure 2), resulting in a size of $200 \sim 500$ megapixels. Also, as all slabs from the biopsy are very similar, doctors decided which one will be used .

Next, we ask pathologists for labeling tumorous zones in the biopsy slide, distinguishing affected biopsies from healthy ones, and locating the affected areas (like Figure 2).

However, even using only one slab from the biopsy, the image is too big to be used as such. To cope with this problem, we followed the approach of cropping the image in many rectangular tiles, treating the image as many tiles on a wall. We did this inspired on previous works in the medicine field [3]. Making this, we could process each tile independently from others, solving the size problem. As a downside, we missed some relevant information such as the relative position of the file in the image and the surrounding area.

The slide's background had much noise. Our first approach was to try to relax the white's definition. Then, we considered that every pixel with each RGB component above value 215 is white, instead of 255. After that, it was necessary to define how many white pixels should have the tile to be marked as white. We studied our database to find a good percentage (Figure 3). We selected 3 values (95%, 50%, and 20%), and test if there was any difference. As it can be seen in the Figure 3, 95% was the best value.

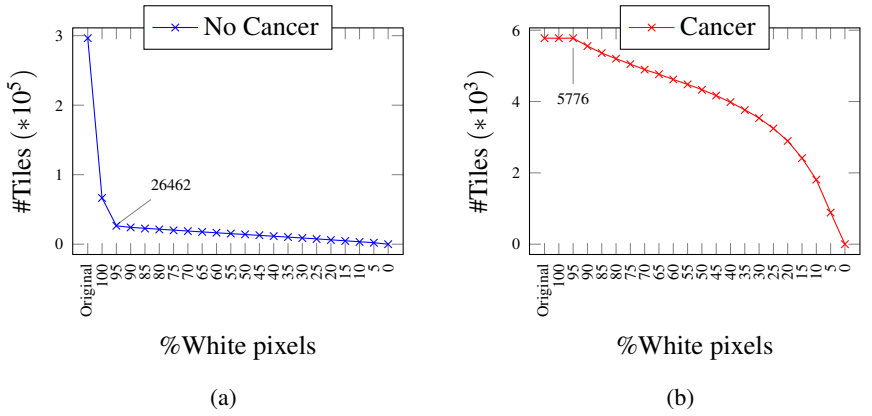


Figure 3. Number of tiles left. **a**: amount of no cancer tiles in our database after removing tiles with x% of white pixels. **b**: amount of cancer tiles in our database after removing tiles with x% of white pixels.

The next question we faced was the following: How should be the proportion of cancerous tiles against healthy tiles? To answer this, we prepared 3 datasets with different proportions: 30% cancer - 70% no cancer, 50% cancer - 50% no cancer, and 70% cancer - 30% no cancer. We found that, in our case, giving enough epochs, all datasets got approximately the same precision. Another problem when 50% - 50% is not used is the class imbalance which can force the network to favour classes most common in the database, sometimes to the limit of outputting always that class.

As images are between 0 and 255 in all of their components, we finished the preprocessing step adding a normalization stage, with the objective of mapping all values between 0 and 1. In this case, this stage divides all the components by 255.

When working with small datasets, a common practice is data augmentation. There are a lot of data augmentation techniques, from rotating and flipping to brightness and contrast changes. We started using random rotations and flips, achieving a noticeable improvement in the accuracy of the network.

3.3. Neural Network

As mentioned, on other studies, there are plenty of different neural networks, such as ResNet [20], Inception, Alexnet, VCG, LeNet, etc. In this case, we decided to test two networks (Figure 1). The first one is a small convolution neural network inspired in LeNet. And the second one is Inception V3, very used in medical image analysis. Initially, we thought that the smaller network would be faster and more accurate because it could specialize more than larger one.

Transfer learning is very popular today, especially when the amount of available data is very low. This technique allows to use a good starting point and requiring less epoch to learn the problem. However, in this work, we wanted to start from the beginning. As we do not used it, all weights were randomly initialized.

3.4. Postprocessing

From the last layer of the network (SoftMax), we got two probabilities, the first one was the probability of being a healthy tile, and the second was the probability of being a

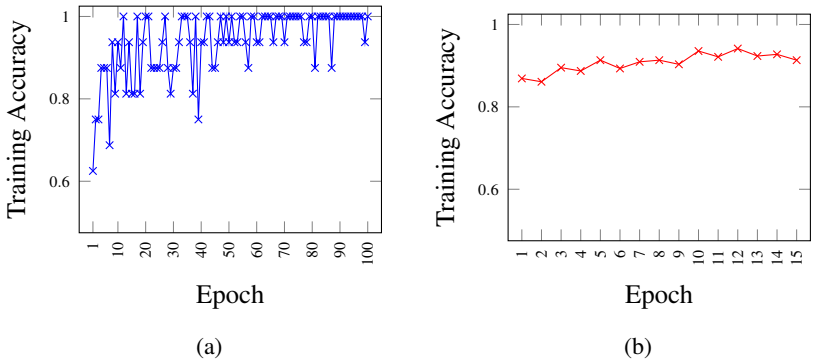


Figure 4. Training accuracy per epoch. **a:** LeNet based network with 100 epochs. **b:** Inception V3 network with 15 epochs.

tumorous tile. Then, we took a threshold to determine when an image is cancerous and when it was not, in this case we used 50% as the threshold. Later, we started to use alpha blending to get a heatmap.

Then, the output gives the probability for each tile of being healthy or not. However, this is not enough. In the clinical environment, doctors want to see the output in a more visual manner. We proposed two ways to approach this: masking and recoloring. Masking implies making an image to be superimposed to the original, allowing to see both images at the same time. Recoloring is similar to the mask but applied directly to the image. Both methods were very similar but imply different results. For this study, we chose recolor, because it allowed us to have only one image at the output and not carrying both.

4. Results & Lessons Learned

In the training phase, we run our two neural network models with random rotations and flips, a learning rate of 0.001, and 0.9 as momentum. All values initialized with Xavier average at 1. The LeNet based network was run for a total of 100 epochs with 8 images per batch. And Inception V3 for 15 epochs with 16 images per batch.

During this work, obtaining data was a very complex task, and we ended using 21 prostate biopsies (391,174 tiles). From that, 17 was for training (302,186 tiles), and after cleaning and data normalization, we ended with 11,552 tiles. These numbers could seem quite large, but they are from 17 biopsies. Therefore, in Figure 4, we can observe that LeNet network is overfitted by the small input data.

In the testing phase, we used 4 biopsies (88,988 tiles), and after cleaning and data normalization, we ended with 7,954 tiles. We achieved an AUC of 82% using Inception V3, and an AUC of 65% in our small network. We show an example tissue, the ground truth by pathologists, and the result obtained from the network reconstructed (Figure 5).

Regarding to Table 1 and Figure 5, our small network was able to address a little about the recognition of tumorous tiles in a prostate biopsy in a reasonable time. Also the inference of a new biopsy is really fast. On the other hand, Inception V3 takes the double of time making only 15% of epochs, but achieving better results. The main problem was the shortage of the data, because our data did not gather all kind of tumorous biopsies.

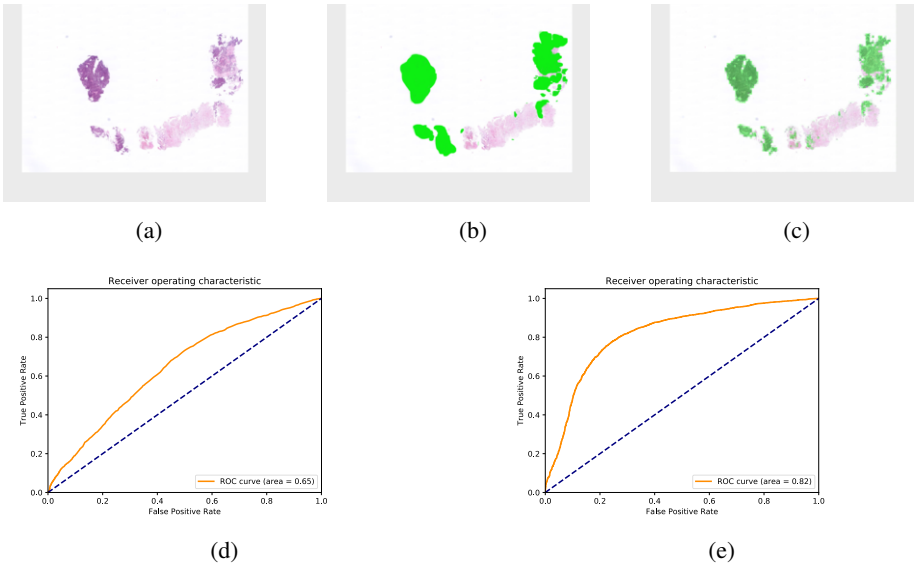


Figure 5. Obtained Results: **a**: Example Tissue, **b**: Ground Truth, **c**: Output Image. ROC Curves with AUC value (**d**: for own LeNet based network, **e**: Inception V3).

Task	Time
Initial Preprocessing	15 mins
Data Base	1 min
Data sieve	3 hours
Postprocessing	18 mins

(a)

Phases	Networks	
	LeNet	V3
Training	1 hour	2 hours
Inference	2 min	6 min

(b)

Table 1. Execution Times: a: Times for common processing tasks in both networks. **b:** Time for each network phase.

From all experiences we collected throughout this experiment, we would like to show the main lessons we have learned:

- **The importance of preprocessing:** We started using the raw image and quickly we found that image dimensions and image size were a problem. With the tiling we realized that the network was not able to learn anything, excluding distinguish background from the tissue. Only after making at least one preprocessing technique, we started to get some acceptable results.
- **Quantity and quality of the data:** Neural networks need a lot of data, not all fields of study have data available, and can take too much time to generate them. But letting this aside, the data have to be good data. We need a good sample from all possible values to get good results.
- **Neural network architecture:** There are a lot of pre-made neural networks ready to be trained. We started with a basic convolutional neural network, although soon we tested a more complex neural network. The complex configuration obtained better results, although this made the training phase slower and more memory bound limited.

- **Postprocessing may be critical in some cases:** In real-world applications, the output needs to be understandable. Moreover, in the medical field is mandatory to visualize the output so doctors can check the works done by the deep learning algorithms. Therefore, in this sector is important both to obtain a good accuracy and to properly show the output in a clear and user-friendly way.
- **HPC requirements:** The training phase is very expensive in computational power, due to tthe many calculations made to crunch the big data used. It is really easy to have memory boundary problems when working with neural networks. Machines with a reasonable amount of memory and high performance computing help to reduce this phase from months to only some hours.

5. Conclusions & Future Work

In this paper, we have exposed the common problems found when developing a neural network for the first time for a medical case. Starting from the raw data, it is a challenging problem the selection of which kind of data choose and how organize it. Also, there are many parameters to be tuned to start learning patterns from the input. Besides, output format can be relevant and may incur a complex step.

As developing a neural network is a very complex task, we have attempted with this work to show our errors and problems on it to help newcomers. We have concluded that preprocessing and the quantity and quality of the data are very important when looking for good accuracy. Also, our small neural network was outperformed by Inception V3, showing us that the prostate cancer detection could be very complex for our simple network.

Further research could be conducted in various directions. The problem exposed to the neural network may reach a better accuracy obtaining more data and more precise labeling. Also, a more refined neural network and preprocessing steps could help. Additionally, preprocessing and postprocessing techniques could be improved to take less time and get near instant results.

Acknowledgments

We would like to thank Prof. Enrique Poblet-Martinez, Dr. Eduardo Alcaraz-Mateos, and Dr. Francisco Garcia-Molina for obtaining and labeling data, and answering all our questions about this clinical case. Also, we thank Andrés García Meroño for several ideas in image processing. And the rest of laboratory workmates, Francisco Muñoz Martínez, and David Corbalán Navarro for their fruitful discussions. This work was partially funded by the AEI (State Research Agency, Spain) and the ERDF (European Regional Development Fund, EU), under the Contract RTI2018-098156-B-C53.

The data used is not publicly available, but all code and scripts used in this work are available at: https://gitlab.com/OdnetninI/prostate-cancer-detection-using-mxnet-framework.

References

[1] Maciej A. Mazurowski, Mateusz Buda, Ashirbani Saha, and Mustafa R. Bashir. Deep learning in radiology: an overview of the concepts and a survey of the state of the art. *CoRR*, abs/1802.08717, 2018.

[2] Hideharu Ohsugi, Hitoshi Tabuchi, Hiroki Enno, and Naofumi Ishitobi. Accuracy of deep learning, a machine-learning technology, using ultra–wide-field fundus ophthalmoscopy for detecting rhegmatogenous retinal detachment. *Scientific Reports*, 7:9425, 2017.

[3] Angel Cruz-Roa, Hannah Gilmore, Ajay Basavanhally, Michael Feldman, Shridar Ganesan, Natalie N.C. Shih, John Tomaszewski, Fabio A. González, and Anant Madabhushi. Accurate and reproducible invasive breast cancer detection in whole-slide images: A deep learning approach for quantifying tumor extent. *Scientific Reports*, 7(1):46450, 2017.

[4] Nicolas Coudray, Paolo Santiago Ocampo, Theodore Sakellaropoulos, Navneet Narula, Matija Snuderl, David Fenyö, Andre L. Moreira, Narges Razavian, and Aristotelis Tsirigos. Classification and mutation prediction from non–small cell lung cancer histopathology images using deep learning. *Nature Medicine*, 24:1559–1567, 2018.

[5] Andre Esteva, Brett Kuprel, Roberto A. Novoa, Justin Ko, Susan M. Swetter, Helen M. Blau, and Sebastian Thrun. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542(7639):115–118, January 2017.

[6] Christof Angermueller, Tanel Pärnamaa, Leopold Parts, and Oliver Stegle. Deep learning for computational biology. *Molecular Systems Biology*, 12(7):878, 2016.

[7] Kurt Hornik. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2):251 – 257, 1991.

[8] Vivienne Sze, Yu Hsin Chen, Tien Ju Yang, and Joel S. Emer. Efficient processing of deep neural networks: A tutorial and survey. *Proceedings of the IEEE*, 105(12):2295–2329, 2017.

[9] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

[10] Shuxin Zheng, Qi Meng, Taifeng Wang, Wei Chen, Nenghai Yu, Zhiming Ma, and Tie-Yan Liu. Asynchronous stochastic gradient descent with delay compensation for distributed deep learning. *CoRR*, abs/1609.08326, 2016.

[11] Kelly H Zou, A James O'Malley, and Laura Mauri. Receiver-operating characteristic analysis for evaluating diagnostic tests and predictive models. *Circulation*, 115(5):654–657, Feb 2007.

[12] W. G. Hatcher and W. Yu. A survey of deep learning: Platforms, applications and emerging research trends. *IEEE Access*, 6:24411–24432, 2018.

[13] Norman P. Jouppi, Cliff Young, and Nishant et al Patil. In-datacenter performance analysis of a tensor processing unit. In *Proceedings of the 44th Annual International Symposium on Computer Architecture*, ISCA '17, pages 1–12, New York, NY, USA, 2017. ACM.

[14] Simon Rodney, Taimur Tariq Shah, Hitendra RH Patel, and Manit Arya. Key papers in prostate cancer. *Expert Review of Anticancer Therapy*, 14(11):1379–1384, 2014.

[15] Geert Litjens, Clara I. Sánchez, Nadya Timofeeva, Meyke Hermsen, Iris Nagtegaal, Iringo Kovacs, Christina Hulsbergen - van de Kaa, Peter Bult, Bram van Ginneken, and Jeroen van der Laak. Deep learning as a tool for increased accuracy and efficiency of histopathological diagnosis. *Scientific Reports*, 6(1):26286, 2016.

[16] Kunal Nagpal, Davis Foote, and Yun Liu et al. Development and validation of a deep learning algorithm for improving gleason scoring of prostate cancer. *CoRR*, abs/1811.06497, 2018.

[17] Eirini Arvaniti, Kim S. Fricker, Michael Moret, Niels J. Rupp, Thomas Hermanns, Christian Fankhauser, Norbert Wey, Peter J. Wild, Jan H. Rueschoff, and Manfred Claassen. Automated gleason grading of prostate cancer tissue microarrays via deep learning. *bioRxiv*, 2018.

[18] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, Nov 1998.

[19] S. Sarraf and G. Tofighi. Deep learning-based pipeline to recognize alzheimer's disease using fmri data. In *2016 Future Technologies Conference (FTC)*, pages 816–820, Dec 2016.

[20] Songtao Guo and Zhouwang Yang. Multi-channel-resnet: An integration framework towards skin lesion analysis. *Informatics in Medicine Unlocked*, 12:67–74, 2018.