JULIA BEATRIZ YIP

AUTOMATIC DETECTION OF TEST SAMPLE ON IMMUNO HISTOCHEMICAL IMAGES USING DEEP LEARNING TECHNIQUES

(*pre-defense version, compiled at July 15, 2019*)

Trabalho apresentado como requisito parcial à conclusão do Curso de Bacharelado em Informática Biomédica, Setor de Ciências Exatas, da Universidade Federal do Paraná.

Field: *Biomedical Informatics*.

Advisor: David Menotti.

CURITIBA PR - BRAZIL

2019

RESUMO

O câncer de mama é caracterizado pelo crescimento descontrolado do tecido mamário, sendo uma das maiores causas de morte no mundo. O diagnóstico correto e precoce da doença possibilita o tratamento adequado para os pacientes, reduzindo suas chances de morte. É fundamental para a definição do tratamento, a análise da proteína Human Epidermal growth factor Receptor-type 2 (HER-2). As lâminas testadas para a proteína de HER-2 contêm duas amostras de tecido, uma de teste e uma de controle. A amostra de controle é sempre positiva para o teste imuno-histoquímico (IHQ) e não pertence ao mesmo paciente que a amostra de teste, o controle tem papel de calibrar e assegurar a qualidade da análise. Esse trabalho propõe a utilização da rede neural YOLO para localização e classificação da amostra de teste a ser analisado no exame IHQ para diagnosticar a presença do HER-2. Para o treinamento da rede foi usado a base pública de images HistoBC-HER2, que contém imagens histopatologicas coradas tanto com HE quanto com IHQ. Como essa base não possui anotações de *ground truth* para detecção de tecidos, foi necessário uma etapa adicional para criação dessas anotações. O resultado desse trabalho adicionou informações de *ground truth* quanto a detecção de amostras de tecidos da base pública HistoBC-HER2, que poderá ser usado para essa tarefa de detecção. Os experimentos mostraram que cada amostra de tecido possuem diversas características que não são facilmente generalizadas, obtendo um IoU médio de 0,2094 com desvio padrão de 0,2522 ao testar nas imagens IHQ a rede treinada com images HE. Por fim, os resultados obtidos pela YOLO demonstraram que as características aprendidas a partir do treinamento com imagens de HE não são suficientes para detectar as amostras nas imagens de IHQ.

Palavras-chave: Redes Neurais. Detecção de ROIs. Imagens Imuno Histopatológicas.

ABSTRACT

Breast cancer is characterized by an abnormal grown of breast tissue cells, it is considered one of the leading cause of deaths worldwide. The correct and early diagnosis of the disease enables the proper treatment for the patients, reducing the death rates. Analysis of the human epidermal growth factor receptor-2 (HER-2) is fundamental for the determination of the proper treatment. On the IHC microscopic slides where the HER-2 test is performed, there are two tissues samples: test and control tissue samples. The control samples are always HER-2 positive and help to ensure the proper assay performance. Our work propose to use the YOLO neural network to localize and detect the tissue samples to be analyzed on IHC exam. The public dataset HistoBC-HER2 containing both HE and IHC images was used for the network training. Since this dataset did not have ground truth annotations for tissue detection, it was necessary an additional step for manually creating these annotations. The resulting work incremented HistoBC-HER-2 dataset with ground truth annotations for tissue samples detection. Our experiments have shown that each tissue sample has distinctive features that can not be easily generalized. Therefore, we achieved an IoU score of 0.2024 with a standard deviation of 0.2522 when testing on IHC images using the network trained with HE images. Finally, the results from YOLO have demonstrated that the features learned from training the network with HE images were not sufficient to detect samples on IHC images.

Keywords: Neural Networks. ROI Detection. Immuno Histochemical Images

LIST OF FIGURES

LIST OF ACRONYMS

AP	Average Precision
BC	Breat cancer
CAD	Computer-Aided Diagnosis
CNN	Convolutional Neural Network
CPN	Cell Proposal Network
CSA	Circle Scanning Algorithm
CT	Computed Tomography
DDSM	Digital Database for Screening Mammography
DINF	Departamento de Informática
DRIVE	Digital Retinal Images for Vessel Extraction
FN	False Negative
FP	False Positive
GPN	Gaussian Proposal Network
GPU	Graphics Processing Unit
GUI	Graphic User Interface
HAM1000	Human Against Machine with 10000 training images
HE	Hematoxylin and Eosin
HER-2	Human Epidermal Growth Factor Receptor-2
IDRID	Indian Diabetic Retinopathy Image Datase
IHC	Immunohistochemistry
IHQ	Imuno-histoquímico
ILSVRC	ImageNet Large Scale Visual Recognition Challenge
ISBI	International Symposium on Biomedical Imaging
IoU	Intersection Over Union
KL	Kullback-Leibler
MA-DN	Multi-Attribute Descriptor Network
mAP	Mean Average Precision
MIL	Multiple Instance Learning
MSE	Mean Square Error
OD	Optical Disc
TP	True Positive
TN	True Negative
SSD	Single Shot Multibox Detector
R-CNN	Region-based Convolutional Neural Network
RPN	Region Proposal Network

ReLU	Rectifier Linear Unit
ROI	Region of Interest
UFPR	Universidade Federal do Paraná
VOC	Visual Object Classes
VRI	Vision, Robotic and Image Laboratory
WSI	Whole-Slide Image
YOLO	You Only Look Once

CONTENTS

1 INTRODUCTION

Cancer is a generic term for a large group of diseases characterized by the growth of abnormal cells beyond their usual boundaries that can then invade adjoining parts of the body and/or spread to other organs [33]. According to Bray et al. [7], cancer is expected to rank as the leading cause of death and the single most important barrier to increasing life expectancy in every country of the world in the 21st century.

In 2018, breast cancer represented 6.6% of all 9.6 million of deaths related to cancer and 11.6% of 18.1 million of new cancer incidences in both sexes. These number are even more significant analyzing only cancer among women, where breast cancer corresponds to 15.0% and 24.2% of deaths and new incidences respectively [7]. Furthermore, breast cancer was the most common type of cancer among woman in most of the countries, including in Brazil, as seen in Figure 1.1.

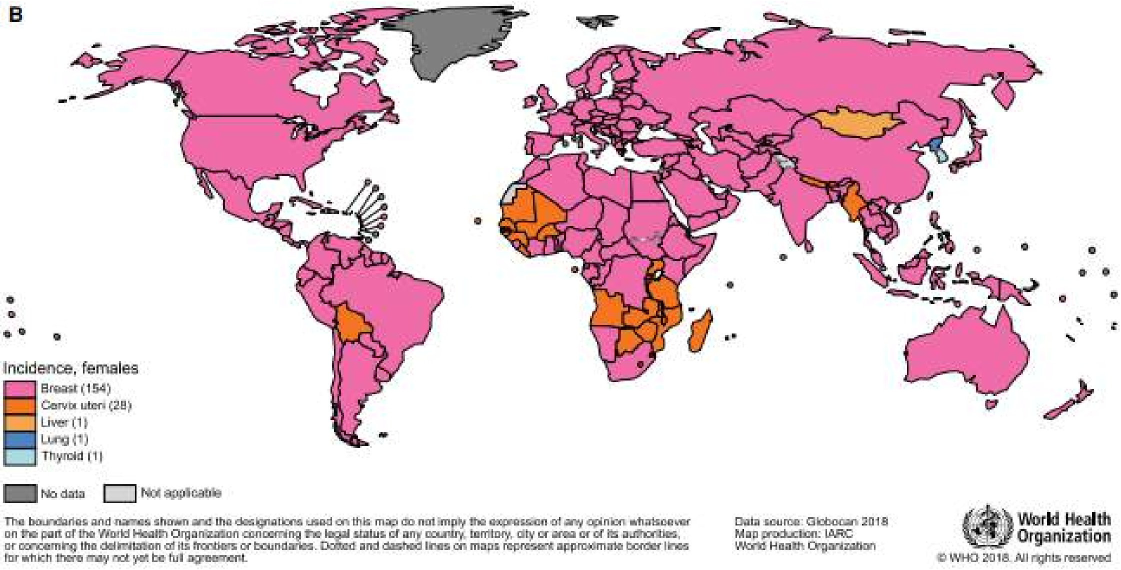
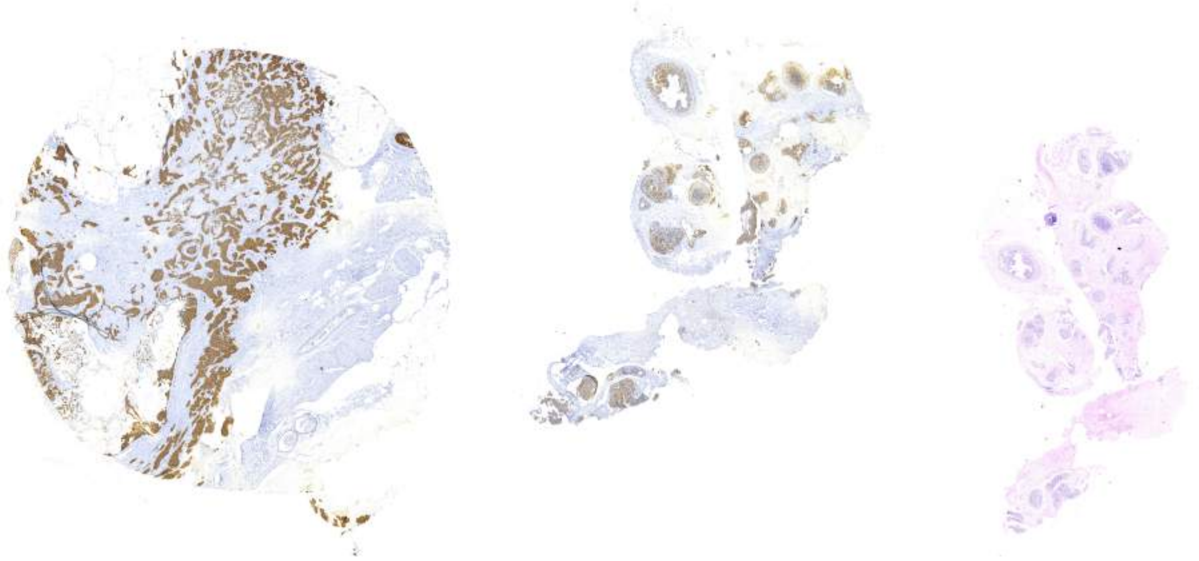

B
Incidence, females
Breast (154)
Cervix uteri (28)
Liver (1)
Lung (1)
Thyroid (1)
No data
Not applicable
The boundaries and names shown and the designations used on this map do not imply the expression of any opinion whatsoever on the part of the World Health Organization concerning the legal status of any country, territory, city or area or of its authorities, or concerning the delimitation of its frontiers or boundaries. Dotted and dashed lines on maps represent approximate border lines for which there may not yet be full agreement.
Data source: Globocan 2018
Map production: IARC
World Health Organization
World Health Organization
© WHO 2018. All rights reserved

Figure 1.1: Global map presenting the most common type of cancer incidence in 2018 in each country among women. The cancer type and the number of countries represented in each ranking group are included in the legend. Figure reproduced from Bray et al. [7].

A correct and early diagnosis of the disease enables the proper treatment for the patients, increasing breast cancer cure rates [33]. For as much as not all chemotherapeutics have the same effect on all breast cancer patients, immunohistochemistry (IHC) has an important role to define which breast cancer subtype is affecting the patient to address the treatment correctly Zaha [51].

One of the protein overexpressions identified on IHC analysis is the human epidermal growth factor receptor-2 (HER-2). The HER-2 positive breast cancer represents a special subtype that has clear epidemiological, clinical, molecular and prognostic differences. A positive HER-2 breast cancer is considered separate entity with recognized worse prognosis and poor response to conventional chemotherapy agents alone Gonzalo Jr Recondo et al. [15].

On the IHC microscopic slides where the HER-2 test is performed, there are two tissues samples: test and control tissue samples. The control samples are always HER-2 positive and do not belong to the patient, who is having his tissue sample tested. These controls help to ensure proper assay performance and to calibrate the appropriate assay sensitivity and dynamic range for each staining run. Furthermore, they also ensure that the reagents are performing properly Hicks and Schiffhauer [20]. To identify correctly which sample contains the test sample, it is common in the pathology routine a presence of a hematoxylin and eosin (HE) stained slide along with its correspondent IHC slide. The HE slide consists of subsequent cuts from the same tissue that is going to be examined in the IHC slide, i.e. the HE slide has tissues samples with similar shape as the IHC, as represented in Figure 1.2.



(a) IHC slide

(b) HE slide

Figure 1.2: An example from HistoBC-HER2 dataset Cordeiro [8]

Due to the availability of microscopic slide scanners and computer aided diagnosis (CAD) systems, computerized image analysis in histopathology of breast tumors holds promise to improve breast cancer diagnosis Robertson et al. [41]. The large volume of data acquired with this scanners makes the manual evaluation inefficient or even impossible, hence microscopic image analysis has gained attention since these method can improve efficiency and objectiveness Lu et al. [31].

However, a typical histopathology slide contains a tissue area of approximately 15×15 mm, which at the resolutions that the slides are scanned, one slide image can require a storage up to gigabytes Veta et al. [48], Robertson et al. [41], Webster and Dunstan [50]. Additionally to the computational problems that can be caused when processing these large images, a considerable part of those images are empty. Thus, it is common to manually identify regions of the slides to perform further analysis.

The selection of region of interest (ROI) on the input images manually performed by a pathologist reduces the computational time and avoid unnecessary processing. Nevertheless, it makes the algorithm dependent of user interaction Jian et al. [23]. By reducing this requirement, we might be closer to a fully-automatic image processing algorithm.

Motivated by these reasons, we propose a method to localize tissue samples on IHC and classify them as test if the detected sample is similar to the tissue sample on the correspondent HE slide. The presented work is going to avoid unnecessary processing on empty spaces, reducing computational time. It also provides a step forward to a fully-automatic image processing algorithm.

1.1 OBJECTIVES

Our aim in the presented work is to detect test tissue samples on histopathological images. We employed the state-of-art You Only Look Once (YOLO) neural network in this tissue detection task. To achieve this general objective, we define the following specific objectives:

- Analyze stained histopathological images. This analysis is necessary to create ground-truth bounding boxes in the dataset from Cordeiro [8];
- Train YOLO with the dataset from [8];
- Obtain the bounding boxes coordinates found by the neural network;
- If there are more than one bounding box, determine which one is going to be evaluated;
- Evaluate the tissue detection task performed by YOLO.

1.2 DOCUMENT STRUCTURE

The presented document is structured in 6 chapters. This first chapter gives an introduction about the breast cancer problem in the world health. It also it emphasizes the importance of early and correct diagnosis followed by a proper treatment. Moreover, it describes our motivation and defines our general and specific objectives.

Chapter 2 introduces concepts used in the development of this work. It describes the metrics used, fundamental neural network theory and specific concepts of the YOLO neural network.

The following chapter contains a brief overview of previous works that approached similar problems.

Chapter 4 explains the methodology employed in this work. This chapter also details the dataset used.

The results of our experiments along with their analyzes are found in chapter 5.

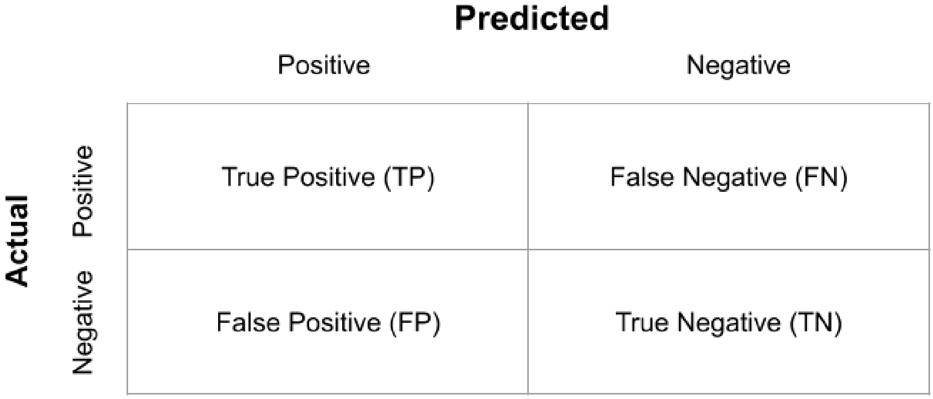
Finally, chapter 6 is the conclusion of this work. It has the final acknowledgements of the problem and future works that might be performed.

2 THEORETICAL FOUNDATION

This chapter explains the theoretical concepts used in the present work. Within this chapter we present commonly evaluation metrics used in object detection problems. We also introduce artificial neural networks, what they are, how they work and the core ideas employed on those algorithms. Lastly, we present YOLO, a state-of-art object detection neural network, and improvements made on its second and third version, i.e. YOLOv2 and YOLOv3.

2.1 EVALUATION METRICS

In machine learning problems, a confusion matrix is used to visualize the algorithm corrected predictions. According to Provost and Kohavi [35], the confusion matrix is a matrix showing the predicted and actual classifications (Figure 2.1).



		Predicted	
		Positive	Negative
Actual	Positive	True Positive (TP)	False Negative (FN)
	Negative	False Positive (FP)	True Negative (TN)

Figure 2.1: Illustration of a confusion matrix.

2.1.1 Metrics in Confusion Matrix

Provost and Kohavi also define the following metrics using the values of a confusing matrix. Within those metrics, we would like to highlight *accuracy*, *recall* and *precision* due to the frequent use of these metrics in object detection context Everingham et al. [12], Lin et al. [30], Russakovsky et al. [43].

Accuracy It is the proportion of corrected predicted examples.

$$accuracy = \frac{TP + TN}{TP + FN + FP + TN}$$

Recall It is the proportion of positive cases that were correctly predicted. It is also known as *true positive rate* or *sensitivity*.

$$recall = \frac{TP}{TP + FN}$$

Specificity It is the proportion of negative cases correctly predicted, also called *true negative rate*.

$$specificity = \frac{TN}{FN + TN}$$

Precision It is the proportion of actual positive cases that were predicted as positive.

$$precision = \frac{TP}{TP + FP}$$

2.1.2 Other Evaluation Metrics

As declared in [12], evaluation of results on multi-class datasets has several problems. Some of those problems are listed bellow.

- An image contains instances of multiple classes, making not trivial an evaluation of the classification task.
- The prior distribution over classes is nonuniform, hence accuracy measure is not appropriated.
- Evaluation metrics need to be algorithm-independent since that different detection algorithms do not share an universal evaluation metric;
- Classification and detection are evaluated as separated tasks.

Therefore, Everingham et al. proposed to use the *average precision* (AP) to evaluate both classification and detection on Pascal Visual Object Classes (VOC) Challenge 2007. The decision was taken due to the observation AP summarises the shape of the precision/recall curve. It is defined as the mean precision for a set of eleven equally spaced recall levels [0, 0.1, ..., 1] (Equation 2.1). The precision at each recall level r is interpolated by taking the maximum precision measured for a method for which the corresponding recall exceeds r (Equation 2.2).

$$AP = \frac{1}{11} \sum_{r \in \{0, 0.1, \ldots, 1\}} p_{interp} r \tag{2.1}$$

$$p_{interp} r = \max_{\tilde{r}:\tilde{r} \geq r} p(\tilde{r}) \tag{2.2}$$

Mean Average Precision Another metric that comes from AP is the mean average precision (mAP), which is the mean value of average precision for each class.

Intersection Over Union (IoU) This metric is the ratio of overlap area between the predicted and ground-truth bounding boxes and their area of union.

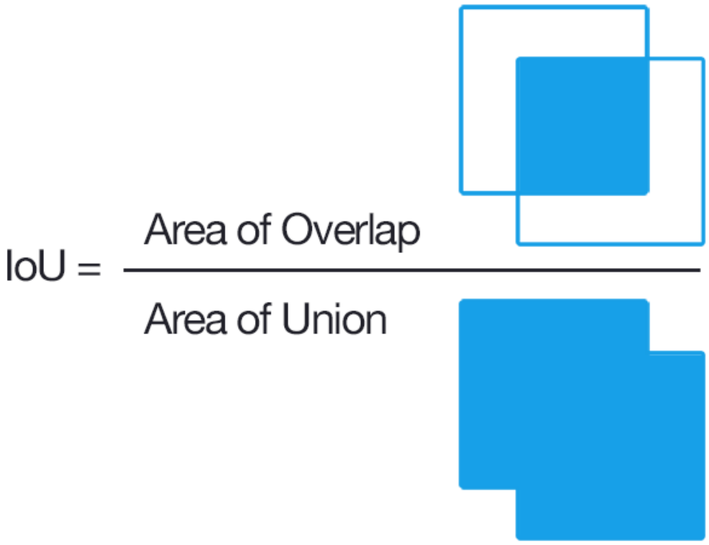


Figure 2.2: Computing the Intersection of Union is as simple as dividing the area of overlap between the bounding boxes by the area of union. Figure reproduced from Rosebrock [42].

2.2 ARTIFICIAL NEURAL NETWORKS

Haykin [17] defines artificial neural network as a machine that is designed to model the way in which the brain performs a particular task or function of interest, as in the brain, the knowledge in the neural network is acquired through a *learning process*, this process consists in a massive interconnection of the basic cell in the brain, the neurons. Corresponding to this learning process that happens biologically, artificial neural networks mimic those mechanisms through a net of neurons cells that perform simple calculations and are interconnected with synaptic weights.

The properties and capabilities of artificial neural networks make them suitable to be used in an extensive variety of problems. Firstly, it is important to emphasize its parallel distributed system and, secondly, its ability to learn and to find a reasonable output corresponding to an input that was not encountered in the training process. The Figure 2.3 represents a dataset with elements illustrated by points in the three graphs. Each graph has one curve that tries to model those elements. There is a curve that has a *underfitting* of the dataset (left), a curve that has a *good fitting* (middle) and a curve is *overfitting* (right). Whilst, the left graph represents a curve too simple to model data, the right graph represents a curve too complex that memorized all the training examples, neither left nor right curves represent a model with a good *generalization* of the problem.

Some other crucial properties highlighted by Haykin are:

Nonlinearity Neural networks can be either linear or nonlinear. This characteristic is highly important when comes to model nonlinear problems.

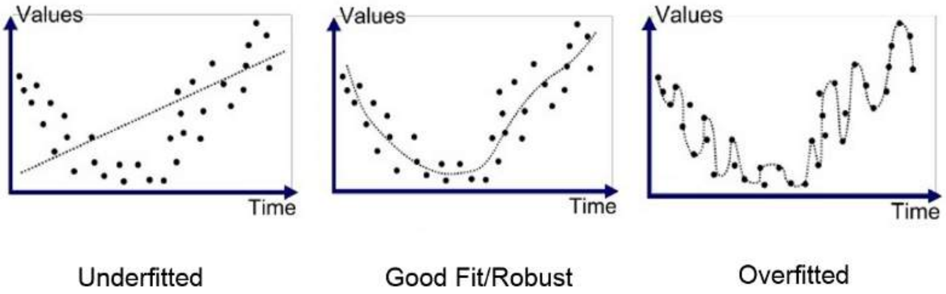


Figure 2.3: The figure shows three curves that are underfitting, with proper fitting and overfitting the problem dataset. Figure reproduced from [6]

Input-output mapping As the category of *supervised learning*, a neural network learns with examples that have a correspondent output for the given input. This correspondence is achieved by an input-output mapping.

Adaptability In a nonstationary environment, i.e. when statistics change with time, neural networks have the capability of adapting its synaptic weights to the changes, thus they may be easily retrained.

Evidential Response In pattern classification problems, additionally to the information about the pattern selected, neural networks also give the confidence score in which the selection was made being a helpful resource to improve the classification.

Contextual Information The knowledge is represented by the structure and activity state of the network. Therefore, every neuron of the network is potentially affected by the global activities of other neurons.

2.2.1 Neuron Model

As mention before, the fundamental information-processing unit of a neural network is called *neuron*. The neuron is a simple calculation cell that is composed of *synaptic weights*, that numerically express the importance of the interconnections represented; an *adder*, that performs a linear combination summing the weighted inputs; an *activation function*, that limits the amplitude of the output; and optionally a *bias*, which makes an adjustment in the input of the activation function depending whether it is positive or negative.

Inspired by Warren McCulloch's and Walter Pitts's early work, Frank Rosenblatt developed between 1950 and 1960 a type of artificial neural network called *perceptron* Nielsen [32]. Perceptron is the simplest neural network with only one neuron. The comprehension of a perceptron behavior is fundamental to understand more complex neural networks that are more popular nowadays. Moreover, it will be helpful in the explanation of elements of a neural network.

Elements and Notation To illustrate the elements of a neuron labeled k, the model in Figure 2.4 is going to be used. The neuron receives a set of *inputs* and generates an *output*. This result is obtained by executing an activation function applied to the weighted sum of the inputs increased by the neuron bias.

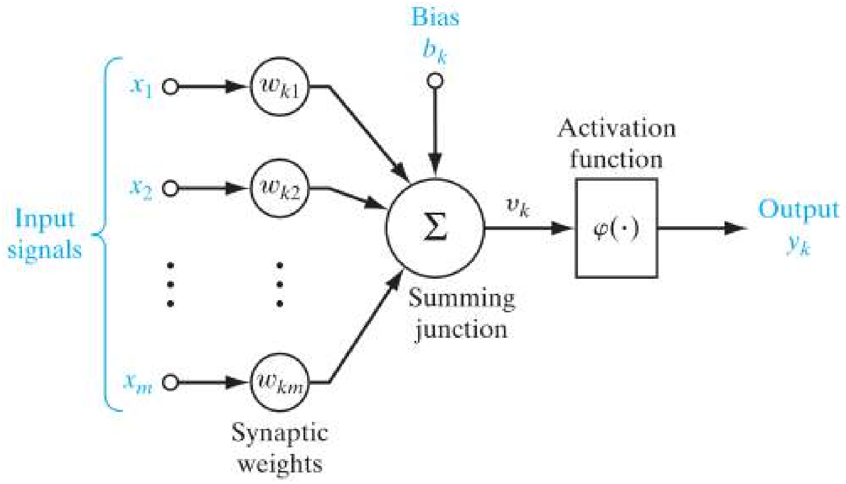


Figure 2.4: Model of a neuron, labeled k. Figure reproduced from [17].

- Inputs: the input signals that the network receives are simply called *inputs* and usually are listed as x_1, x_2 and so goes on until the last signal, x_m.
- Output: the output signal from neuron is denoted as y_k.
- Weights: synaptic weights are denotes as w_{kj}, where j indicates the input while the k indicates the neuron that are interconnected by this weight.
- Bias: b_k represents the bias of neuron k.
- Activation Function: $\phi(\cdot)$ expresses the activation function chosen.

With this notation, the output of a neuron k can be expressed with Equation 2.3. Alternatively, $\sum_{j=1}^{k=m} w_{kj}x_j$ can be rewrite as a dot product of vectors w_k and x_k, whose elements represent the weights and inputs respectively, $\sum_{j=1}^{k=m} w_{kj}x_j \equiv w_k \cdot x$, Equation 2.4 is the corresponding equation for this notation.

$$y_k = \phi(\sum_{j=1}^{k=m} w_{kj}x_j + b_k) \tag{2.3}$$

$$y_k = \phi(w_k \cdot x + b_k) \tag{2.4}$$

2.2.2 Architecture of a Neural Network

A neural network is organized in layers, as seen in Figure 2.5, where the inputs are encoded in neurons and organized in the leftmost layer, the *input layer*. The outputs are encoded and organized in the rightmost layer, the *output layer*. The layers that have neither inputs nor outputs are called *hidden layers*. Moreover, the neural network can have one or multiple hidden layers and the neurons on the second layer can make decisions at a more complex and more abstract level than the ones on first layer, as well as the neurons on the third layer can make decisions even more complex and more abstract than the neurons from the second layer [32].

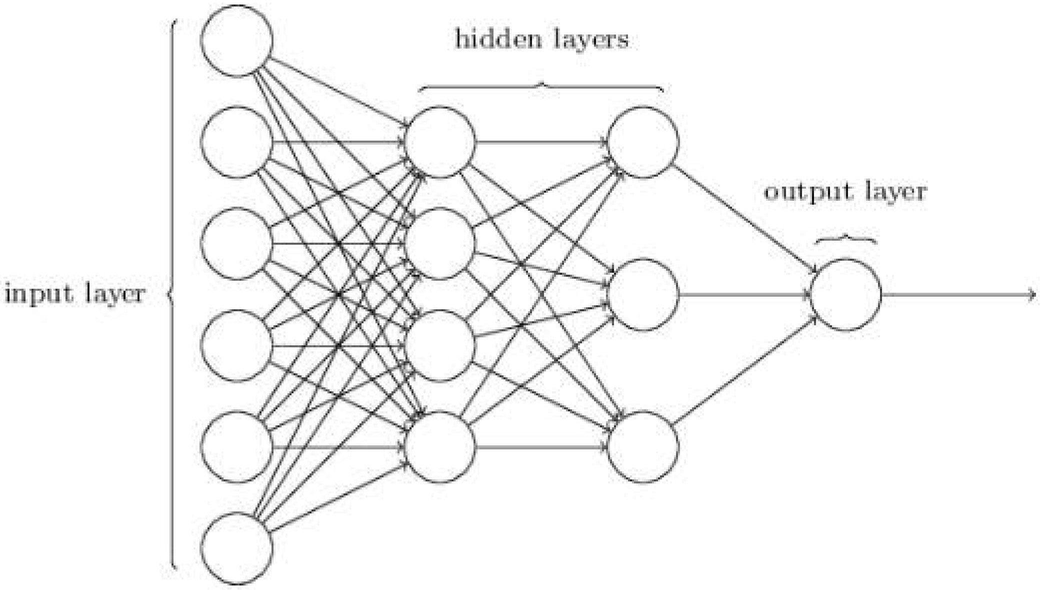


Figure 2.5: An example of a 4-layer neural network with 2 hidden layers. Figure reproduced from [32]

It can be also observed in the Figure 2.5 that the output from one layer is used as input for the next layer. This kind of neural networks where the information is only *fed forward*, i.e. does not have a loop caused by information feed back, is denominated as *feed forward* neural network.

2.2.3 Loss Function

It is desired that the algorithm finds weights and bias where the output approximates from $y(x)$ to the input x. To achieve this goal, it is necessary to evaluate how close the algorithm is from achieving it. The information that quantifies this distance is obtained using the *loss function*. A popular choice of loss function is the *mean square error* (MSE). The MSE function (Equation 2.5) sums the square difference between the correct output y and the predicted $x(\Theta)$.

$$L(\theta) = \frac{1}{N}\sum_{i=1}^{N}(y_i - x_i(\Theta))^2 \tag{2.5}$$

2.2.4 Backpropagation Algorithm

One of the main elements that enable neural networks to learn in the training process, i.e. update its weights and bias according to the error presented by the loss function, is the backpropagation algorithm. It employs two key concepts: the gradient descent and the chain rule. The loss function quantifies the error of the neural network and we aim to find weights and bias that minimizes this error. One method to find the minimum of a function is using the so called *gradient descent*.

At the same time, the *chain rule* is used in the backpropagation algorithm since the output of a neural network can be seen as a result of many composed functions encapsulated in each neuron. The output neuron is computed taking into account the connected neurons from the previous layer with its weights and bias. The neurons of the previous layer are computed in a similar way with respect to their connections with their previous layer and so goes on. Therefore, to accurate how the weights influence the output, the chain rule applied backward (Figure 2.6) is fundamental to minimize the loss of a neural network.

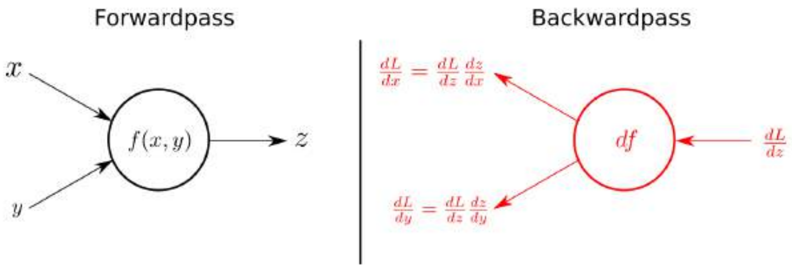

Forwardpass
Backwardpass
x
y
$f(x,y)$
z
$\frac{dL}{dx} = \frac{dL}{dz}\frac{dz}{dx}$
df
$\frac{dL}{dz}$
$\frac{dL}{dy} = \frac{dL}{dz}\frac{dz}{dy}$

Figure 2.6: The forward pass on the left in calculates z as a function $f(x, y)$ using the input variables x and y. The right side of the figures shows the backward pass. Receiving $\frac{dL}{dz}$, the gradient of the loss function with respect to z from above, the gradients of x and y on the loss function can be calculate by applying the chain rule. Figure reproduced from Kratzert [25].

2.3 DEEP NEURAL NETWORKS

As defined by LeCun et al. [27], deep learning methods are representation-learning methods with multiple levels of representation, that are obtained by composing simple but non-linear modules. Each module transforms the representation at one level (starting with the raw input) into a representation at a higher, slightly more abstract level. Furthermore, LeCun et al. also claim that deep learning allows computational models that are composed of multiple processing layers to learn representations of data with multiple levels of abstraction. Those models receive the name of *Deep Neural Networks*. This is now achievable due to the the advent of fast graphics processing units (GPUs).

Since 2009, deep neural networks have won many official international pattern recognition competitions [44]. One remarkable achievement in this area was in 2012, when a deep convolutional network architecture *AlexNet* [26] won ImageNet Large Scale Visual Recognition Challenge 2012 (ILSVRC2012) and this type of architecture became popular in image processing.

Recently, deep learning is emerging as a powerful tool and will continue to attract considerable interests in microscopy image analysis including nucleus detection, cell segmentation, extraction of regions of interest (ROIs), image classification, etc. [31].

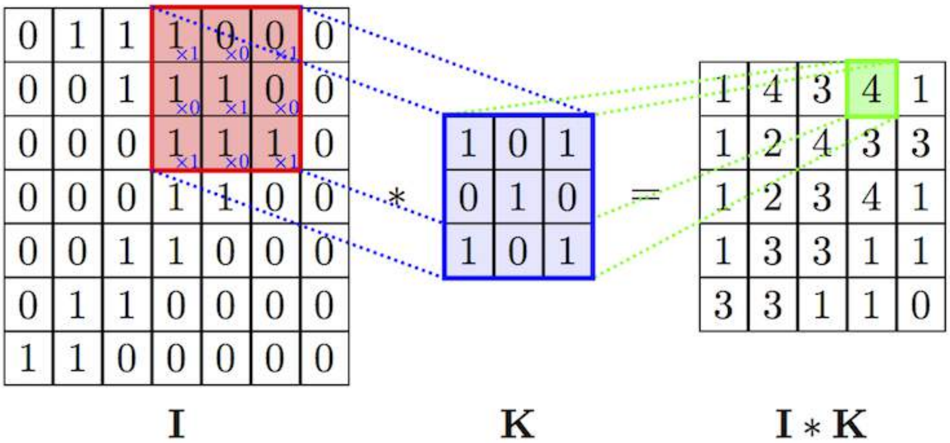
2.3.1 Deep Convolutional Neural Networks

Convolutional neural networks or simply ConvNets are a category of artificial neural networks that are composed of multiple layers of neurons. These architectures are formed by stacked convolutional, pooling and fully-connected layers, along with activation functions [27].

The use of many layers, its local connectivity, shared parameters, and pooling can be listed as core ideas of ConvNets; these properties enable an easier training process at the same time that maintains a considerable accuracy [27].

Following the deep learning, deep convolutional networks have gained attention in the last years. This success came from the efficient use of GPUs, new activation functions as Rectifier Linear Unit (ReLU), a new regularization technique called dropout, and techniques to generate more training examples by deforming the existing ones [27].

Convolution Layer The convolutional layers combine information of two functions (I and K) in an operation that combines them to produce a third one, as represented by the image 2.7. In the network, I is our input image, K is the convolution kernel, i.e. our parameters, which are going to be learned in the training process and the result $I * K$ is the feature map [9].



$$\begin{matrix}0&1&1&1_{\times 1}&0_{\times 0}&0_{\times 1}&0\\0&0&1&1_{\times 0}&1_{\times 1}&0_{\times 0}&0\\0&0&0&1_{\times 1}&1_{\times 0}&1_{\times 1}&0\\0&0&0&1&1&0&0\\0&0&1&1&0&0&0\\0&1&1&0&0&0&0\\1&1&0&0&0&0&0\end{matrix} * \begin{matrix}1&0&1\\0&1&0\\1&0&1\end{matrix} = \begin{matrix}1&4&3&4&1\\1&2&4&3&3\\1&2&3&4&1\\1&3&3&1&1\\3&3&1&1&0\end{matrix}$$

Figure 2.7: Illustration of a convolution operation in the image I using the convolution kernel K. Figure reproduced from [45].

Pooling Layer The pooling operation reduces the dimensionality through downsampling in width and in height of the given input, and thus reducing the number of parameters to be trained, the training time and the probability of overfitting. Downsampling can be obtained using the maximum, mean, median or other reduction operations in the values of the selected window [49].

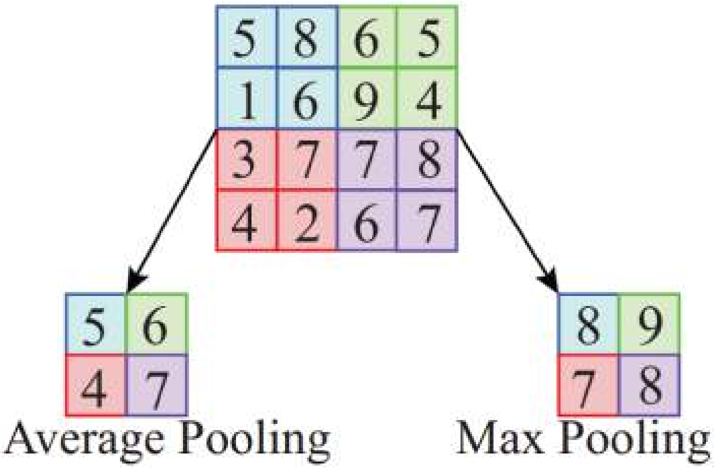


Figure 2.8: Representation of a max and a average pooling operation with a 2x2 window and stride 2 in a 4x4 input. Figure reproduced from [22]

Fully-connected Layer The fully-connected layers are those whose neurons have connections to the neurons on the previous layer. Similarly to a convolutional layer, a fully-connect layer performs a matrix multiplication with its layer weights followed by a bias offset in the neural network context [32].

Activation Function The activation function includes non-linearity to our model, which is important when modeling generic information. Popular choices are the Sigmoid, Tanh, Rectifier Linear Unit (ReLU) and SoftMax function [32, 36].

Dropout Layer A popular regularization method to reduce overfitting in Convolutional Neural Networks is the insertion of Dropout Layers. During the training phase, some neurons with a random probability are dropped, i.e. they are not going to be considered in the feedforward neither backpropagation, as illustrated in Figure 2.9.

2.4 OPTIMIZATION ON DEEP NEURAL NETWORKS

Complementing the studies of deep neural networks over the recent years, some works have been done in improving the accuracy of neural networks, speeding up the training process and reducing overfitting. Examples of the optimization techniques on deep neural networks are the batch normalization, introduced by Ioffe and Szegedy [21], and data augmentation techniques discussed on Perez and Wang [34].

2.4.1 Batch Normalization

Ioffe and Szegedy [21] have shown an improvement in the network training if a batch normalization is introduced after each convolution. It is possible to list some advantages after this batch

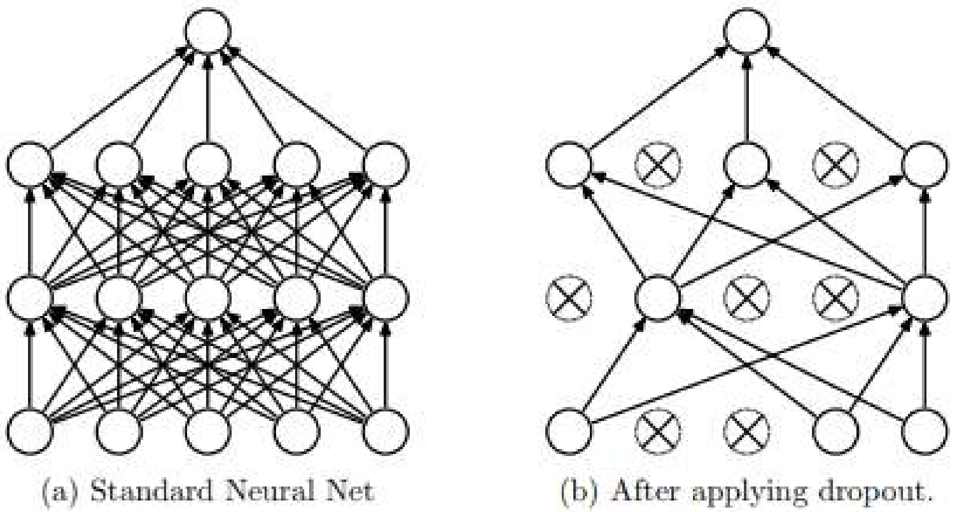


Figure 2.9: The network on the left is an example of standard neural network with 2 layers and the network on right is an example with dropout. Figure reproduced from [46].

normalization process, as the possibility of using higher learning and fewer worries about weights initialization.

The input values x are normalized by the batch mean μ and the batch variance σ. Optionally the result can be scaled by γ and added to an offset β as shown in Equation 2.6.

$$\frac{\gamma(x-\mu)}{\sigma}+\beta \tag{2.6}$$

2.4.2 Data Augmentation

As the results of a neural network rely heavily upon examples encountered during the training process, it is desired that those examples cover a wide range of coverage for the problem that the network aims to solve. This is particularly important when working with medical images, once that the data usually is not as vast as other categories of images. Thus limiting the size of the dataset can lead to overfitting [34].

Artificial examples can be generated in a procedure known as *data augmentation*, which applies image transformations on the dataset, e.g. flipping and rotation [34].

2.5 YOU ONLY LOOK ONCE NEURAL NETWORK

A popular approach in object detection is the region proposal-based technique, which creates candidates windows that are in sequel classified using a convolutional neural network features [14]. Another approach to object detection problem is a system based on mixtures of multiscale deformable part models which makes use of a sliding window technique [13].

Different from sliding window and region proposal-based techniques, You Only Look Once Neural Network, also called YOLO, is a state-of-art neural network for object detection. It sees the entire image during training and test time so it implicitly encodes contextual information about classes as well as their appearances [40] (Figure 2.10).

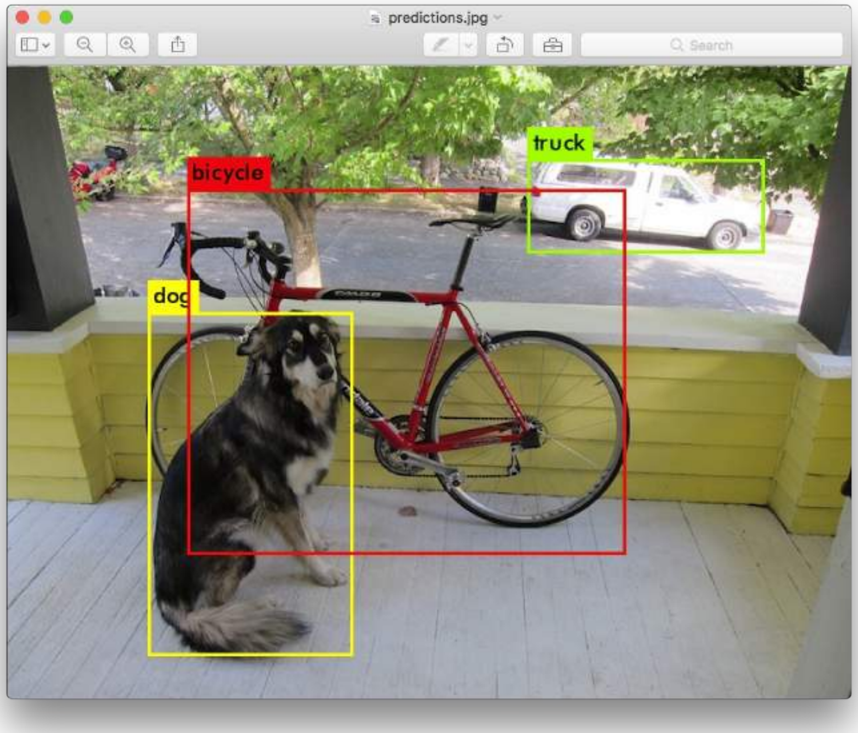


Figure 2.10: Illustration of the final prediction using YOLO. Figure reproduced from [38].

As stated by Redmon et al., YOLO is fast, it reasons globally about the image when making predictions and learns generalized representations of objects. The elements introduced originally in [40], further improved in [37] and also [39] are described in the following subchapters.

2.5.1 Unified Detection

Redmon et al. [40] have unified the elements from an object detection system, i.e. object localization and object classification, into one neural network. These neural networks use features from the entire image to predict multiple bounding boxes and their probabilities.

YOLO divides the input image into a $S \times S$ grid. Each grid cell is responsible for the object that it is localized in the center of the cell. Additionally, each grid cell predicts B (limit number to be defined in the problem statement) bounding boxes and their confidence scores. The confidence score is a quantification of how confident the model is about the prediction. Its values is calculated as $Pr(Object) \star IOU_{truth}^{pred}$. If there are no objects in that cell, the confidence is zero, otherwise, the confidence is the intersection over the union of the ground-truth and predicted bounding boxes [40].

Each bounding box is defined by 5 predictions: x, y, w, h and the confidence score defined above. The pair of coordinates (x, y) represents the center, while w is the *width* and h

is the *height* of the bounding box respectively. Those predictions can be either relative to the bounds of the grid cell (x and y) or relative to the whole image dimensions (w and h) [40].

The grid cell also predicts C conditional class probabilities $Pr(class_i|object)$ (Equation 2.7). It is important that these probabilities are conditioned on the grid cell containing an object and only one set of classes probabilities is predicted per grid cell. Finally, Figure 2.11 summarizes YOLO object detection system [40].

$$Pr(class_i|object) * Pr(object) * IoU_{pred}^{truth} = Pr(class_i) * IoU_{pred}^{truth} \tag{2.7}$$

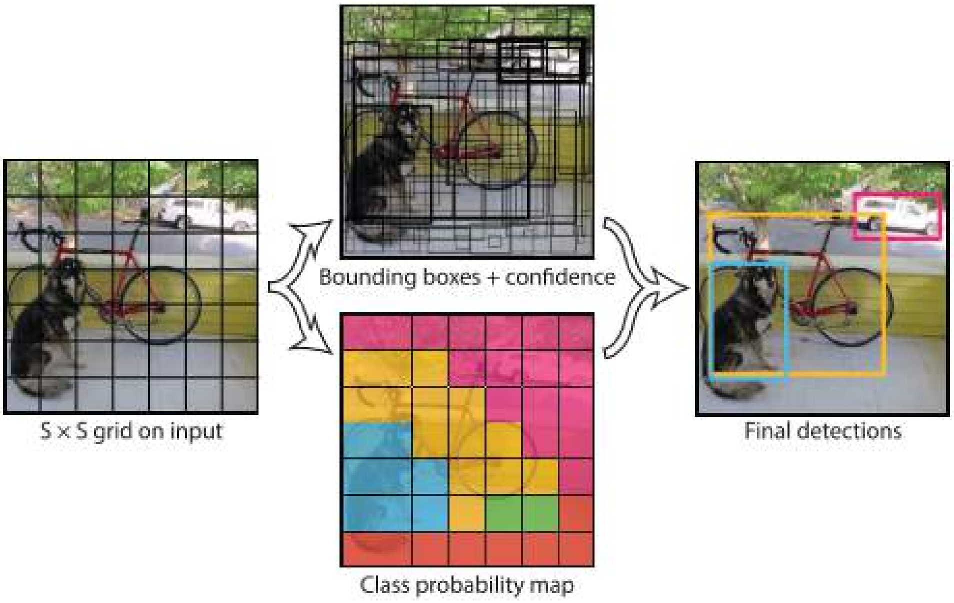


Figure 2.11: YOLO divides the image into an $S \times S$ grid and for each grid cell predicts B bounding boxes, confidence for those boxes, and C class probabilities. Figure reproduced from [40].

2.5.2 Network Design

The network architecture presented by Redmon et al. was inspired by the GoogLeNet model for image classification [40]. However, the inception modules were replaced by 1×1 reduction layers, i.e. 1×1 convolutional layers that have effect on reducing feature dimensions, followed by 3×3 convolutional layers.

The YOLO detection network has 24 convolutional layers followed by 2 fully connected layers. Alternating 1×1 convolutional layers reduce the features space from preceding layers. YOLO has the convolutional layers pre-trained on the ImageNet classification task at half the resolution (224×224 input image) and then double the resolution for detection full network as described by Redmon et al. [40] (Figure 2.12).

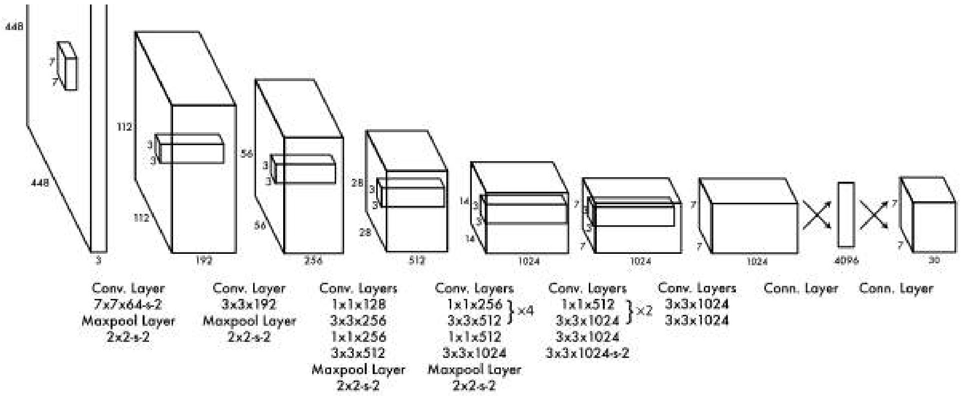


Figure 2.12: The full network. Illustration reproduced from [40].

2.5.3 Modifications in v2

In YOLOv2 [37] features were fine-tuned to improve the network performance. We can highlight the following improvements in YOLOv2 [37]:

Batch Normalization The addition of batch normalization in all convolutional layers in YOLO leads to significant improvements in convergence. It also eliminated the dependency for other forms of regularization.

High-Resolution Classifier The original YOLO trained the classification network with 224 × 224 has increased to 448 × 448 for detection. For YOLOv2, firstly a fine-tune was made on the classification network at the full 448 × 448 resolution for 10 epochs on ImageNet, a state-of-art neural network for image classification [26]. This improvement gives the network time to adjust its filters to work better on higher resolution input. Then YOLOv2 resulting network was fine-tuned on detection.

Convolutional with Anchor Boxes YOLO originally predicted the coordinates of bounding boxes directly using fully-connected layers on top of the convolutional feature extractor. The fully-connected layers from YOLO were removed and *anchor boxes*, also called *priors*, were introduced to predict bounding boxes.

Dimension Clusters Dimension clusters address one of the problems that were encountered in bounding boxes predicted by anchors. Although the network can learn to properly adjust the dimensions of boxes manually picked, starting with better chosen anchors for the network would make easier for the network to learn to predict good detections. Redmon and Farhadi have run k-means clustering on the training set to automatically find better good priors. Instead of using standard k-means with Euclidean distance, $d(box, centroid) = 1 - IoU(box, centroid)$ was used as the distance metric to achieve better IoU scores.

Direct Location Prediction Another problem with the anchor box approach was the model stability, which was addressed using direct location prediction. The network became more stable due to an easier to learn parametrization. This was possible through constraining between 0 and 1 the predicted location coordinates, using logistic activation and relative coordinates to the location of the grid cell.

Multi-Scale Training For a more robust network, every 10 batches the network chooses a new image dimension instead of using a fixed input image size.

Fine-Grained Features In the second version of YOLO, the network predicts on 13×13 feature map. It changes although this change can be sufficient for large objects, fine-grained features might benefit localization of smaller objects.

2.5.4 Modification in v3

Redmon and Farhadi presented a new version of the YOLO neural network. The improvements made in the YOLOv3 are listed below.

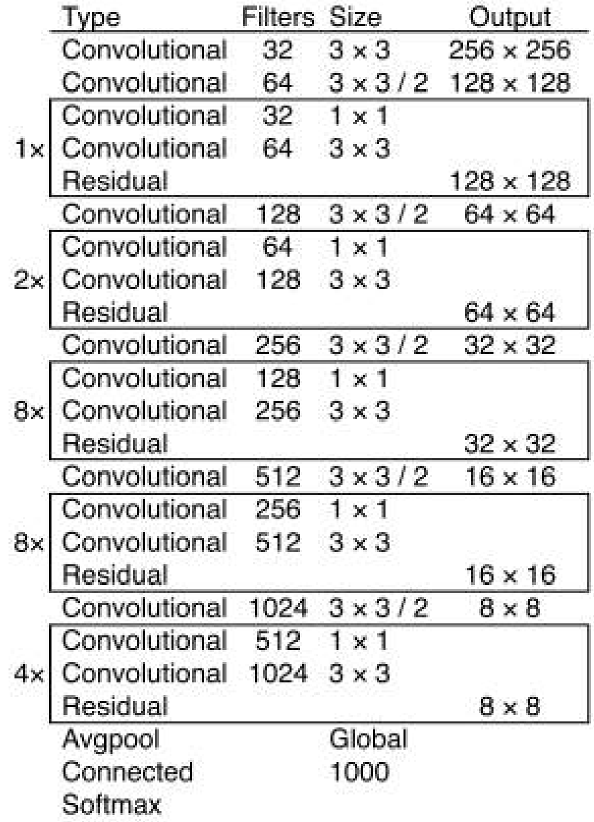
Bounding Box Prediction The third version of YOLO uses logistic regression to predict an objectness score for each bounding box. Moreover, YOLOv3 assign 1 for the objectness score if the bounding box anchor overlaps a ground truth object by more than any other bounding box anchor. Otherwise, if the bounding box anchor is not the best but does overlap a ground truth object by more than some threshold, the prediction is ignored. The YOLO system only attributes one bounding box anchor for each ground truth object.

Class Prediction Multilabel classification is used to predict classes that the bounding box might contain. YOLOv3 uses independent logistic classifiers instead of softmax and binary-cross entropy loss for class training.

Predictions Across Scales Using a similar concept to feature pyramids networks, YOLOv3 predicts boxes at 3 different scales.

Feature Extractor A new network is used to perform feature extraction. The network is a hybrid approach between YOLOv2, Darknet-19 and residual networks. It uses successive 3×3 and 1×1 convolutional layers and shortcut connections as illustrated in Figure 2.13. In total, it has 53 convolutional layers, therefore it received the name Darknet-53.

We choose to use YOLOv3 due to their additional features. The predictions across scales improvement could help to detect tissue samples, considering the variety of sizes and shapes that those objects present. We also decided to use this network, due to the successes of residual networks in previous works on image processing [18].



	Type	Filters	Size	Output
	Convolutional	32	3 × 3	256 × 256
	Convolutional	64	3 × 3 / 2	128 × 128
1×	Convolutional	32	1 × 1	
	Convolutional	64	3 × 3	
	Residual			128 × 128
	Convolutional	128	3 × 3 / 2	64 × 64
2×	Convolutional	64	1 × 1	
	Convolutional	128	3 × 3	
	Residual			64 × 64
	Convolutional	256	3 × 3 / 2	32 × 32
8×	Convolutional	128	1 × 1	
	Convolutional	256	3 × 3	
	Residual			32 × 32
	Convolutional	512	3 × 3 / 2	16 × 16
8×	Convolutional	256	1 × 1	
	Convolutional	512	3 × 3	
	Residual			16 × 16
	Convolutional	1024	3 × 3 / 2	8 × 8
4×	Convolutional	512	1 × 1	
	Convolutional	1024	3 × 3	
	Residual			8 × 8
	Avgpool		Global	
	Connected		1000	
	Softmax			

Figure 2.13: Hybrid network approach between YOLOv2, Darknet-19 and residual networks. Figure reproduced from [39].

3 RELATED WORK

In this chapter, we are going to review the literature that already approached the problem presented in the current work. Firstly, we introduce works to detect Region of Interest (ROI) in medical images in general (Table 3.1). In the sequence, we introduce related work in microscopic histopathological images (Table 3.2).

3.1 DETECTION AND LOCALIZATION OF REGION OF INTEREST ON MEDICAL IMAGES

There are a variety of previous works that approached detection and localization of ROIs problems in medical images. Some examples described below are used in dermatological images Goyal and Yap [16], eye fundus Araújo et al. [5], mammography, ultrasound, X-ray, computed tomography or magnetic resonance imaging Kisilev et al. [24], Eswaraiah and Reddy [11], Li et al. [29], Li [28].

Goyal and Yap proposed to use an InceptionV2 as a base network for feature extractor and classification of anchor boxes with Fast Region-based Convolutional Neural Network (R-CNN) and Single Shot Multibox Detector (SSD) meta-architectures for the ROI lesion detection problem. They used 3 publicly available datasets for skin lesion: International Symposium on Biomedical Imaging 2017 (ISBI-2017) Challenge, PH2, and Human Against Machine with 10000 training images (HAM10000). The method achieved 94.5% of precision and 94.3% of recall for IoU > 0.5 in ISBI-2017, 100% of precision and recall for PH2 and, lastly, 83.3% of precision and 82.4% of recall for HAM1000.

In the field of eye fundus imaging, Araújo et al. proposed a neural network for the task of simultaneous optic disc (OD) detection, fovea detection, and OD segmentation from retinal images. The network denominated UOLO was designed for the detection and segmentation of structures in biomedical images. This network was inspired in the well-established YOLOv2 for the detection task and U-Net for segmentation. For training and testing the network, 3 public eye fundus datasets were used: Messidor; Indian Diabetic Retinopathy Image Dataset (IDRID); and Digital Retinal Images for Vessel Extraction (DRIVE). Araújo et al. achieved equal or better performance in comparison to the state-of-the-art on both detection and segmentation tasks in a single-step prediction, reaching IoU 0.88±0.09 on Messidor dataset.

Kisilev et al. proposed a multi-task convolutional neural network (CNN) approach for detection and semantic description of lesions in diagnostic images. Therefore network is trained to perform 2 tasks: to detect ROI candidates and to generate semantic descriptions of the lesions inside the ROIs. In the first task, Region Proposal Network (RPN) module is employed to generate ROI candidates. The module is also trained to predict ROI bounding box coordinates and its scores. While in the second task, the candidates ROIs are used as input for a Multi-Attribute

Descriptor Network (MA-DN) in a multi-class-multi-label prediction problem. Kisilev et al. used the public Digital Database for Screening Mammography (DDSM) and proprietary ultrasound images. The approach used ROI detection as part of their pipeline, thus only evaluating the overall method.

X-ray images usually have multiple sites of abnormalities with monotonous and homogeneous image features, making the localization and detection of these abnormalities a non-trivial task. In addition to this problem, there a lack of large-scale datasets with localization annotation for building an accurate prediction model. Thence, Li et al. presented a unified approach that simultaneously improves disease identification and localization with only a small amount of X-ray images containing disease location information. The method firstly employs a convolutional neural network to learn information from the entire image and implicitly encodes both the class and location information for the disease. After that, the image is sliced into a patch grid to capture local information of the disease. Depending on the disease label, the learning task can be either a fully supervised problem or a multiple instance learning problem (MIL). Using NIH Chest X-ray dataset with 14 disease labels for the training, the method proposed by Li et al. outperformed ImageNet pre-trained ResNet-50, which was used as the reference model.

Yet the described methods have approached only rectangular bounding boxes, while lesions characterized by computed tomography (CT) scans are often elliptical. Li proposed a method to detect bounding ellipses to avoid geometry information loss about the lesion regions that might be annotated besides rectangular bounding boxes. The approach uses a Gaussian Proposal Network (GPNs) that learn bounding ellipses as 2D Gaussian distributions on the image plane. The Kullback-Leibler (KL) divergence loss calculated between the proposed and the ground truth Gaussian was used in this extension of Region Proposal Network (RPN). Li used the DeepLesion dataset from NIH, which contains CT slice images. While RPN achieved 31% of sensitivity at the IoU threshold of 0.5, the proposed GPN achieved 36% at the same IoU threshold.

3.2 DETECTION AND LOCALIZATION OF REGION OF INTEREST ON MICROSCOPIC IMAGES

In the field of microscopic imaging, several methods have been proposed for the detection and localization of ROIs as well. There are approaches for detecting fluorescent HeLa cells as Akram et al. [1], cancer cells detection in phase-contrast images as Zhang et al. [52], detection of ROIs using graphic user interface (GUI) tool for compression as Dong et al. [10] and also histopathological tissue detection as Alomari et al. [3, 4].

To overcome problems in fluorescent microscopic images, as the low contrast, variable fluorescence, weak boundaries, conjoined and overlapping cells, Akram et al. proposed a CNN that perform cell detection, segmentation and tracking. In this method, Akram et al. have proposed a cell proposal network (CPN) based on fully convolutional neural network architectures. The

CPN consists on *feat* part for an extraction of 256-dimensional feature vector, followed by two parallel fully connected layers, *score* and *bbox*. While **bbox** part predict the bounding box coordinates, the *score* predict the score for this bounding box. The method achieved an average precision of 96.3% on the Fluo-N2DL-HeLa dataset from the International Symposium of Biomedical Imaging (ISBI) cell tracking challenge.

In phase-contrast microscopy images, Zhang et al. have proposed a deep detector for cancer cells using Faster Region-Convolutional Neural Network (R-CNN). The Faster R-CNN is applied for cell detection, later a Circle Scanning Algorithm (CSA) is used for further re-detection of adhesion cancer cells. A private dataset was used for the experiments, resulting in a better performance using the Faster R-CNN combined with CSA than only using the Faster R-CNN on both adhesion cell area and all cell area.

For malaria-infected cell detection in whole slide images, Dong et al. presented a graphical user interface (GUI) tool which allows the user to control the ROI extraction and evaluate compression algorithms on ROIs generated. Using the GUI tool proposed, a ROI can be extracted using a block matching algorithm from a template given by the user.

As for histopathological images, Alomari et al. have proposed a hybrid tissue segmentation method that combines a reduced feature vector using PCA and a neural network with hidden layers. The method achieved an accuracy of 96.9%, 94% and 96.5% in the training, validation, and test set respectively. Similarly, Hiary et al. [19] have proposed a method to localize and segment tissues using unsupervised learning techniques as clustering. It achieved accuracy of 96.6%, 95.3% and 96.0% in training, validation and test set. The main advantage of this method compared to the method presented in Alomari et al. [3] is that the clustering-based technique does not need training. Although achieving good results depends on the seeds initialization, i.e. the initial labels given to each cluster.

Reference	Year	Author	Problem	Method	Dataset	Results
[24]	2011	Kisilev et al.	ROI lesion detection and semantic description	RPN + MA-DN	DDSM (mammography)	only evaluated semantic description of the detected ROIs
					proprietary dataset (ultrasound)	
[16]	2018	Goyal and Yap	skin lesion ROI detection	InceptV2 + R-CNN + SSD	ISBI-2017	precision 94.5%; recall 94.5%
					PH2	precision 100%; recall 100%
					HAM1000	precision 83.3%; recall 82.4%
[5]	2018	Araújo et al.	OD detection and segmentation; fovea detection	YOLOv2 + U-Net	Messidor	0.88±0.09 IoU
					IDRID DRIVE	
[29]	2018	Li et al.	X-ray images abnormalities detection	CNN + MIL	NIH Chest X-ray	accuracy 0.01 ± 0.00 for Nodule (IoU > 0.7) accuracy 0.98 ± 0.02 for Cardiomegaly (IoU > 0.1)
[28]	2019	Li	bounding ellipses detection on CT images	GPN	DeepLesion (NIH)	sensitivity 36% (IoU 0.5)

Table 3.1: ROI detection and localization related work on medical images

Reference	Year	Author	Problem	Method	Dataset	Results
[1]	2016	Akram et al.	cell detection, segmentation and tracking	CPN	Fluo-N2DL-HeLa	precision 96.3%
[52]	2016	Zhang et al.	cell detection	Faster R-CNN + CSA	private dataset	precision 0.996; recall 0.9
[10]	2016	Dong et al.	ROI cell detection and compression algorithms on ROI	template	private dataset	only evaluates compression
[3]	2009	Alomari et al.	tissue localization	PCA + 2 hidden layer NN	private dataset	accuracy of 96.9% (training), 94% (validation) and 96.5% (test)
[19]	2013	Hiary et al.	tissue localization	K-means	private dataset	accuracy of 96.6% (training), 95.3% (validation) and 96% (test)

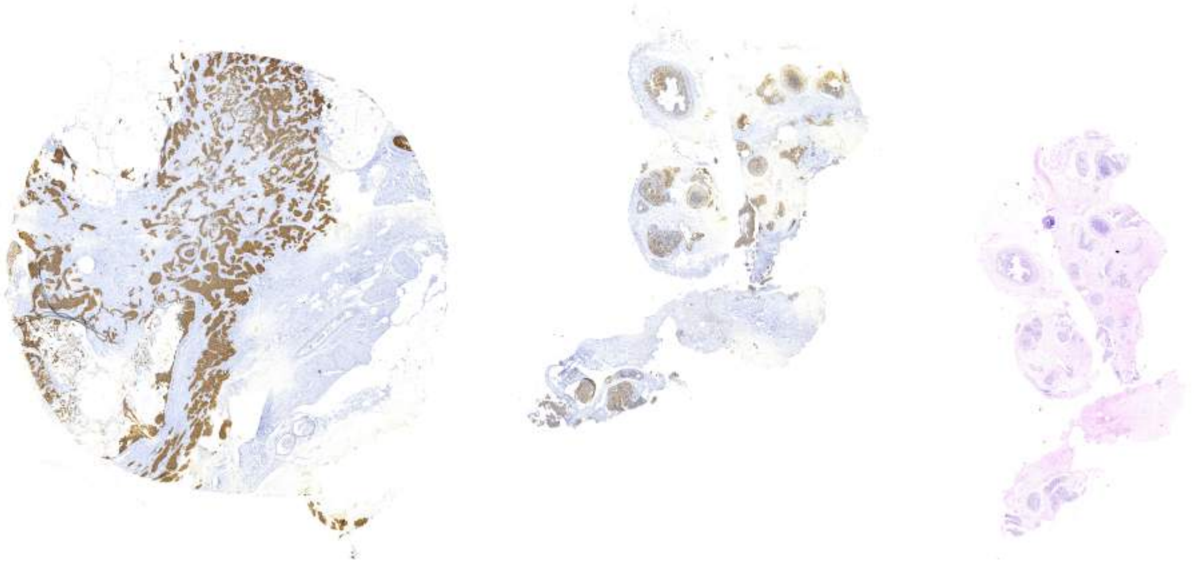
Table 3.2: ROI detection and localization related work on microscopic images

4 MATERIALS AND METHODS

In this chapter, we describe the dataset used in our work, as well as the data augmentation that was used. Further, we also discuss the implementation of the network, its configurations and the hardware on which the experiments were performed.

4.1 DATASET

The presented work used the public HistoBC-HER2 dataset, created by Cordeiro [8]. The HistoBC-HER2 dataset has 140 WSI images from breast cancer tissues. Each WSI contains an IHC and its correspondent HE slide as represented in Figure 4.1.



(a) IHC slide

(b) HE slide

Figure 4.1: An example from HistoBC-HER2 dataset [8]

This dataset was scanned in a Zeiss Axio Scan Z1 with the objective lens of x40, therefore the images are saved in the proprietary *.czi* file format by default. Cordeiro also has provided the images in *.jpeg* format, which was used as the network input. This image format was used due to the network framework used did not support *.czi* files.

4.2 GROUND TRUTH ANNOTATION

Since the HistoBC-HER2 dataset did not provide ground truth for bounding boxes, it was necessary to manually mark them using labelImage [47]. LabelImage is a graphical image annotation tool and label object bounding boxes in images, which saves the annotations in a *.txt* file of either PascalVOC or YOLO format.

The annotations of the bounding boxes ground truth for the HistoBC-HER2 dataset were further analyzed by Cordeiro.

4.3 HARDWARE

The experiments were performed at Vision, Robotic and Image (VRI) Laboratory from the University Federal of Paraná. To perform the experiments it was used a GPU from Nvidia GeForce RTX 2080 Ti series from one of VRI servers. The server has 16 processors from Intel(R) Core(TM) i9-9900K CPU @3.60GHz series and total RAM memory of 62GB.

4.4 IMPLEMENTATION

There are YOLO implementations available on code hosting platforms, e.g. GitHub [1]. It was to chosen to use the version from AlexeyAB [2] considering its well documented repository.

4.4.1 Data Augmentation

To improve the network performance and avoid overfitting, the implementation [2] allows changing the data augmentation ratio in the network configuration file. The possible hyper-parameters, i.e. parameters that are set before the training process, are listed below.

- If saturation=1.5 then it will be changes saturation= init_value $*$ rand(1/1.5, 1.5).
- If exposure=1.5 then it will be changes exposure= init_value $*$ rand(1/1.5, 1.5).
- If hue=0.1 then it will be changes hue= init_value + rand(-0.1, 0.1).
- If angle=30 the the image will be randomly rotated by +30 or -30 degrees.

Since the data is randomly augmented, we do not have an image of how many images were artificially generated using those hyper-parameters.

The set of data augmentation hyper-parameters used can be found in the network configuration file [2].

4.4.2 Training

For the training process, we split the dataset using the ratio of 80% for training and 20% for test. We assured that the same patient was not on both training and test folders.

The YOLOv3 was at first trained using both HE and IHC images from the training set to learn the general features of both stained tissues. AlexeyAB recommends to train for

[1] https://github.com/

[2] https://github.com/julbyip/yolov3-tissuedetection

$2000 \times classes$, as we aim to identify test and control classes, we changed the hyper-parameter $max\_batches$ for 5000 to have a margin of 1000 extra batches.

Then, for each pair of HE and IHC images, we retrained the YOLOv3 with the weights that achieved the best average IoU in the previous step. In this second training, we used only the HE image from one patient in each training process. For the second training, we set additional 200 batches over the batch number that achieved the best IoU ($max\_batches = best\_batch + 200$).

In the end, we had 1 trained YOLOv3 network for the general features of histopathological images and 1 trained YOLOv3 network for each patient on the test set.

We set *width=960*, *high=960*, *subdivisions=64* hyper-parameters to improve performance as recommended by AlexeyAB. For other hyper-parameters as *momentum*, *decay*, *learning_rate* and others we used the default values [3].

The original YOLov3 *.cfg* configuration file uses width=416 and height=416 for input dimensions. Besides, the configuration file set batch=64, and subdivisions=16, i.e. a batch of 24 images is used for every training step and each batch is further divided by 15 to decrease GPU requirements. The original *.jpeg* images have width=4001 and height=5000, hence automatic resizing of the image would lead to significant information loss. Empirically, we came to the hyper-parameters width=960, height=960, batches=64, and subdivisions=64 to avoid information loss caused by the resizing meanwhile respecting memory limitations. We aimed to choose values multiples of 32 as recommended by AlexeyAB.

[3] `https://github.com/julbyip/yolov3-tissuedetection`

5 RESULTS AND DISCUSSION

This chapter presents and discusses the experimental results that were obtained using the material and methods described previously.

5.1 EXPERIMENTAL RESULTS

For the final evaluation, we tested the YOLOv3 model with weights from different epochs for each patient on the test set and compared the resulting IoUs.

We used the weights from the epoch which achieved the best average IoU over all the HE images from the test set. The best epoch in our evaluation using HE images was 3760 as represented in Figure 5.1.

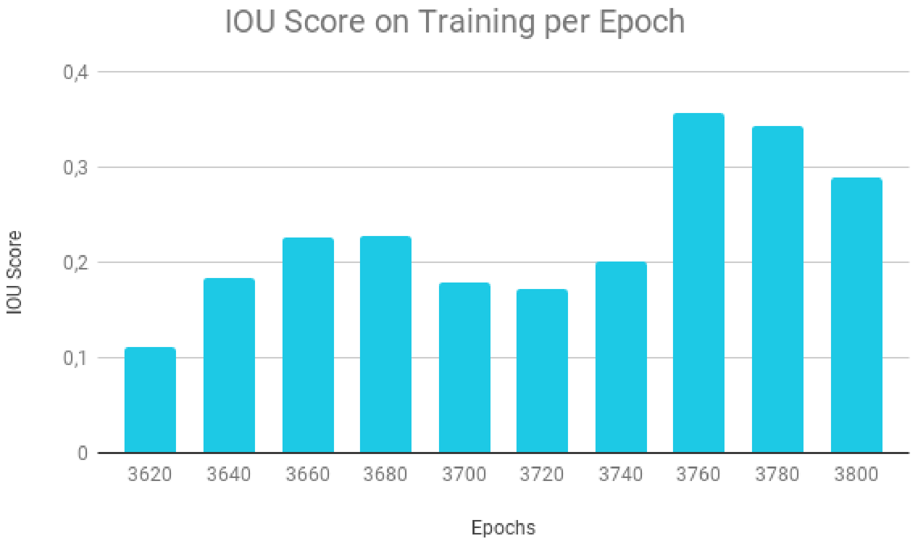


Figure 5.1: Graph with average IoU scores per epoch on HE images.

Then, we tested the IHC images on with the weights from this epoch. In some images, the network recognized more than 1 bounding box as represented in Figure 5.2. When this happened, we chose the bounding box with the highest confidence score.

Figure 5.3 represents the IoU scores per patient when testing YOLOv3 for the IHC images using the weights found on YOLOv3 training with HE images. At last, we achieved 0.2094 IoU score employing the method described in this work, with a standard deviation of 0.2522.

In our experiments, we had also tried to use the weights from the best epoch for each patient. However, in this approach the network only recognized tissue samples on 2 of 29 patients from the test set, resulting into an IoU score of 0.02855 and a standard deviation of 0.1078.

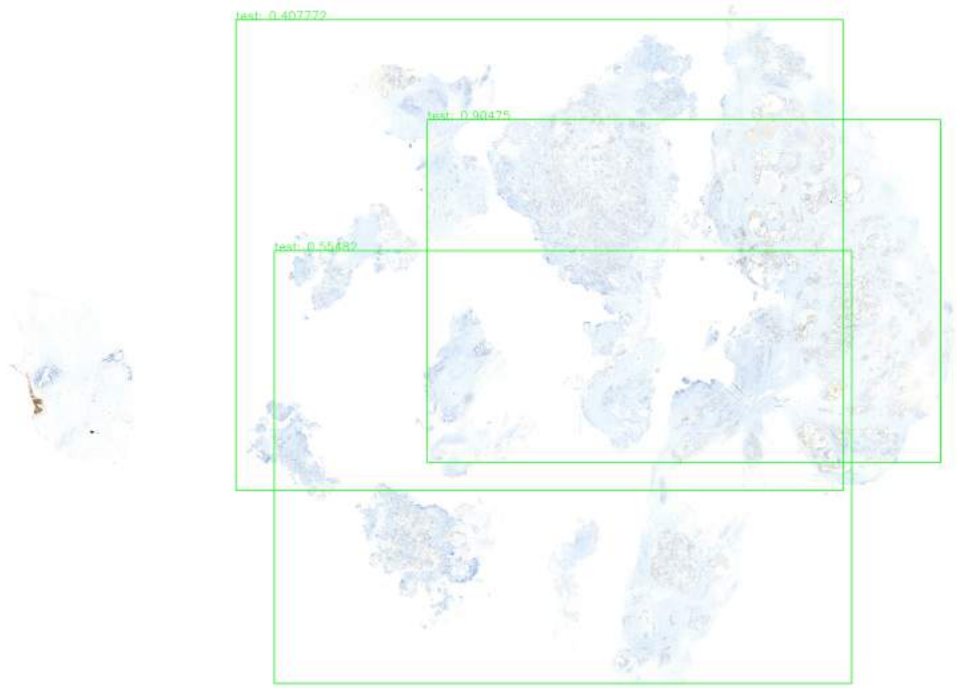


Figure 5.2: Example of a IHC image with multiple bounding boxes detected.

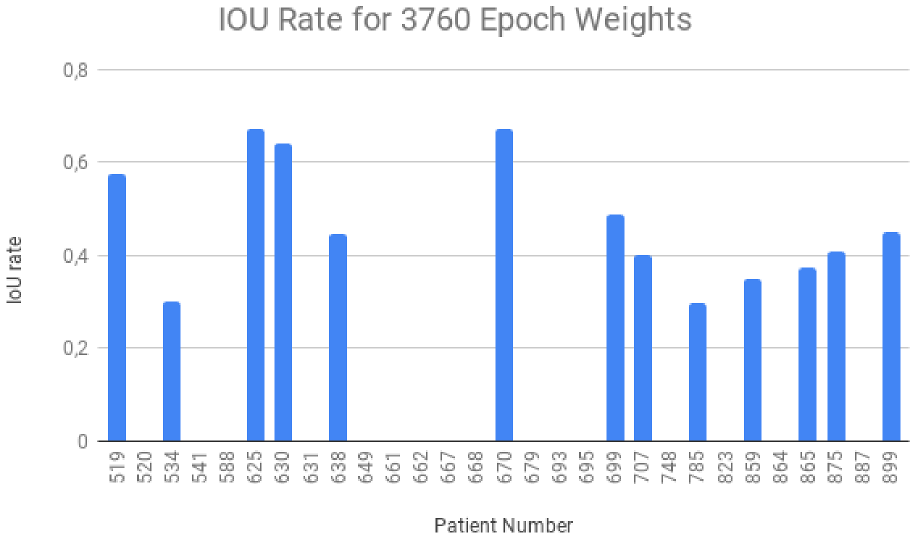


Figure 5.3: Graph with IoU scores for all the patients using the weights from the best epoch.

Additionally, we tested on IHC images using the mode best epoch, which was 3780. On this approach, we achieved an IoU score of 0.1963 with a standard deviation of 0.2702.

5.2 DISCUSSION

Tissue sample detection had shown a not trivial task since these samples have a wide range of shapes. It was not easy to infer a general pattern for the test sample on IHC slides since it depends on the analysis of a secondary slide. Although the HE slide has tissue samples with a similar shape, its different staining can produce an image with different color and texture features. Therefore, our experiments have shown that although shape features can be used to detect test samples on IHC slides, color and texture feature have a decisive role in the detection task.

Figure 5.4 compares the average IoU score per patient overall epochs saved. We can see in this Figure that tissue samples from patients 631, 649, 661, 662, 679 and 887 were not able to be recognized in any epoch, therefore averaging 0.0 on IoU score. While patient 630, 638 and 899 achieved more than 0.35 on overall IoU score.

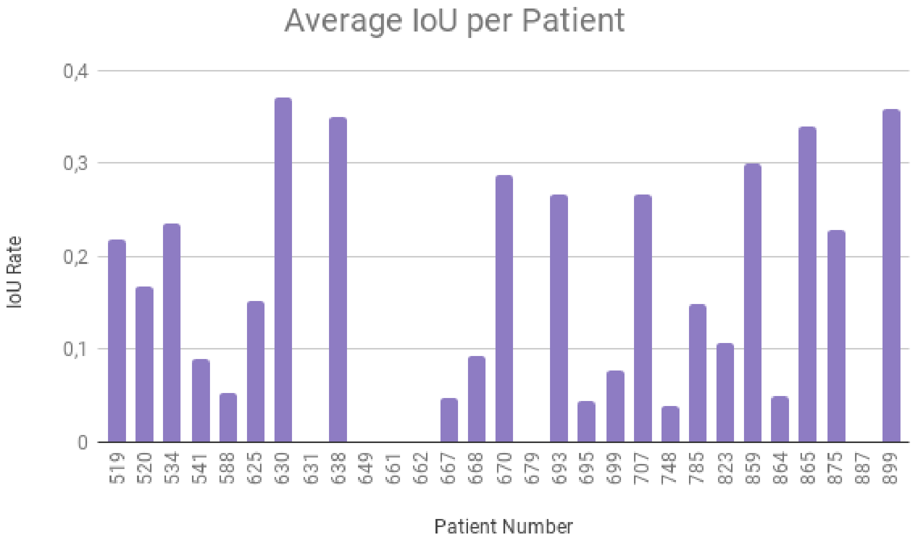
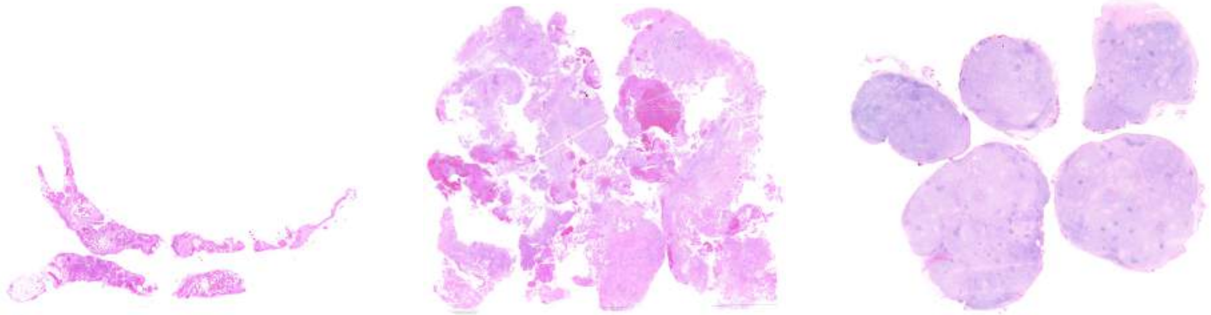


Figure 5.4: Graph from average IoU score per patient over all epochs

Analyzing the slides images from those patients that were successfully recognized by the trained YOLOv3 (Figure 5.5 and Figure 5.6), it is possible to infer some similarities. On those examples, both HE and IHQ slides are properly stained and we can see clearly the tissue borders.



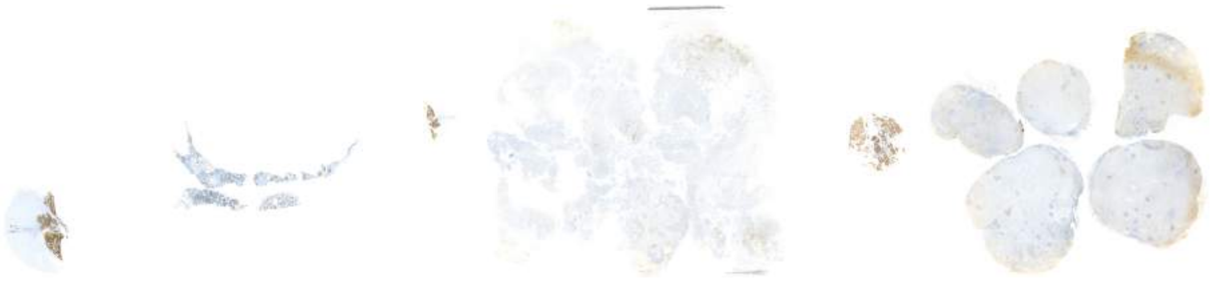
(a) Patient 630

(b) Patient 638

(c) Patient 899

Figure 5.5: Examples of HE images from patients were able to be recognized by the trained YOLOv3.

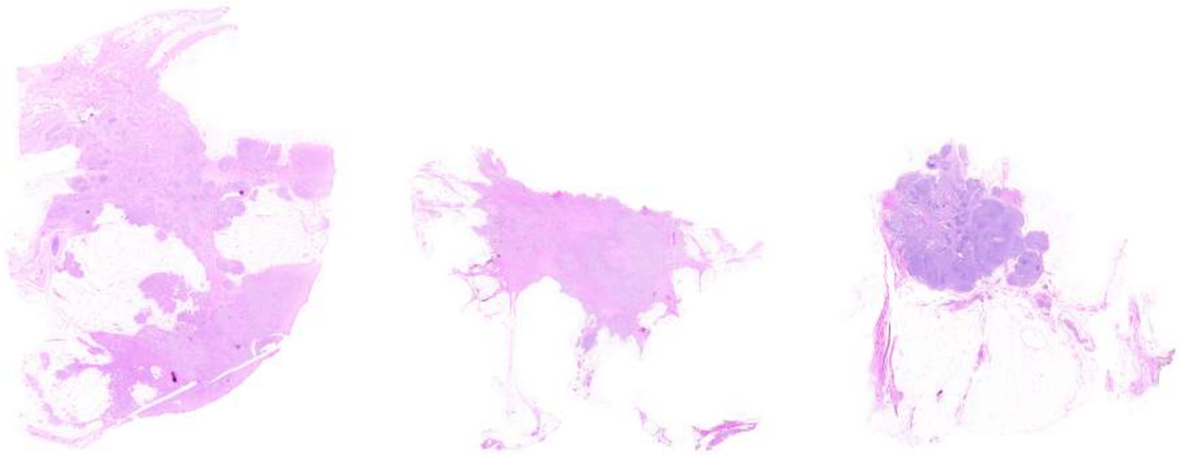
Figure 5.7 and Figure 5.8 are HE and IHC images examples from patients whose tissues samples were not able to be recognized by the trained YOLOv3. The tissue samples on these images have either sparse shape and with irregular contours (Figure 5.7(b) and Figure 5.8(b));



(a) Patient 630 (b) Patient 638 (c) Patient 899

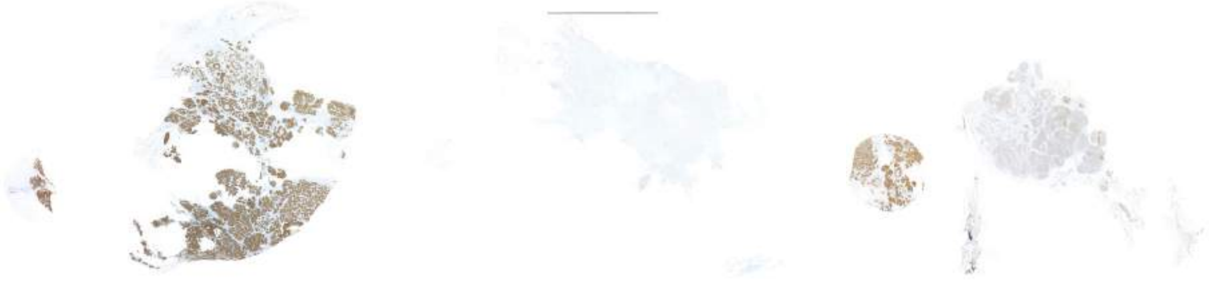
Figure 5.6: Examples of IHC images from patients were able to be recognized by the trained YOLOv3.

either its test was very similar to its control sample (Figure 5.7(a) and Figure 5.8(a)); or the HE and IHC images have differences on their shape(Figure 5.7(c) and Figure 5.8(c)).



(a) Patient 631 (b) Patient 699 (c) Patient 887

Figure 5.7: Examples of HE images from patients were not able to be recognized by the trained YOLOv3.



(a) Patient 631 (b) Patient 699 (c) Patient 887

Figure 5.8: Examples of IHC images from patients were not able to be recognized by the trained YOLOv3.

The same behavior observed on IHC images can be also detected in HE images. Figure 5.9 shows a similar pattern when evaluating the retraining on HE images, the patients that had a low IoU score on IHC usually also have had low IoU score on HE. We can emphasize patient 662 who obtained 0.0 IoU score on both retraining and test.

Moreover, finding the appropriate epoch to stop training before overfitting meanwhile detecting tissue samples is not a simple task. In the experiments, we tried to use the weights from

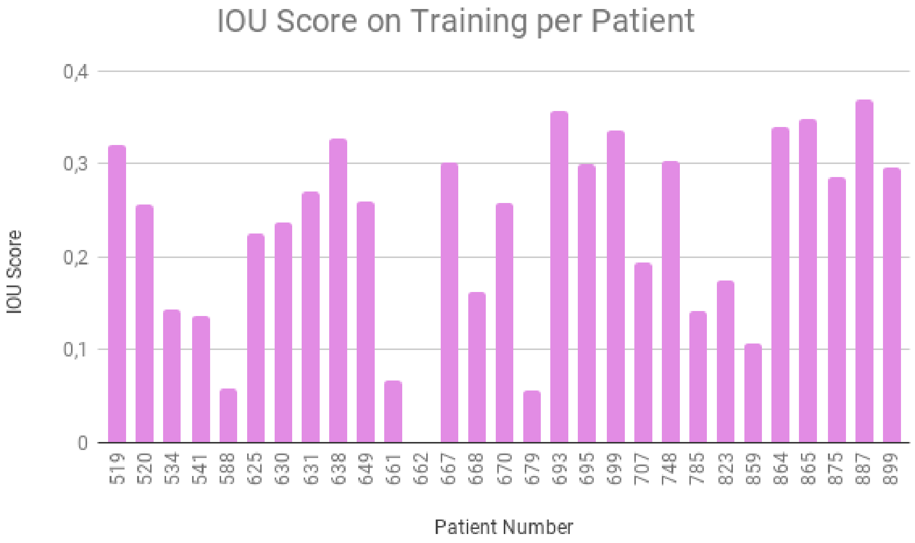


Figure 5.9: Graph from average IoU score per patient over all epochs when evaluating on HE images.

the epoch that achieved the highest IoU score for each patient on HE images. On this approach the network was not able to detect tissue samples on IHC images, implying on overfitting. The retrained YOLOv3 has memorized the features from HE images and could not perform well on IHC images.

Another aspect that could have limited our results are the size of the dataset. As deep learning techniques rely heavily on a large dataset, the HistoBC-HER2 had only images from 140 patients, which is small compared to other datasets used in the training of deep neural networks. We tried to improve the performance using data augmentation hyper-parameters on the network configuration. However, since the data is randomly generated, it is not possible to be sure how many images were actually used. It is important to emphasize that for a neural network an augmented image is different from the original, e.g. the original and its flipped image are considered 2 different images.

Finally, AlexeyAB has stated that there must be at least 1 example of each object in the training set about the same shape, relative size, angle of rotation, tilt, illumination [2]. Additionally, AlexeyAB also recommended having images with objects at different scales, rotations, lightings, from different sides, on different backgrounds. Consequently, the training set should preferably have 2000 different images for each class or more, and the network should train $2000 * classes$ iterations or more.

In our problem, we aimed to label correctly the tissue samples as test, thus we just considered test class. However, considering that usually classes are defined as groups with similar features within the same class, it might be necessary to split the images according to similarities between them and retrain the network with enough examples of each class as recommended.

Nonetheless, it would be necessary to further categorize each tissue sample in one of the most common classes. This approach would be infeasible for the present work, due to the unique appearance of each tissue sample, it is nearly impossible to categorize every sample. Also, the HistoBC-HER2 dataset used in this work does not have enough examples for this approach. Lastly, the manual categorization task is time demanding and beyond the scope of this work.

6 CONCLUSION

The presented work aimed to detect test tissue samples on histopathological images. This step would improve the accuracy of HER-2 computer-aided systems by avoiding processing on empty spaces. Moreover, the method would reduce user dependency for a manual ROI detection.

For this work, we created ground truth annotations for tissue sample detection task on HistoBC-HER2 dataset. These annotations were analyzed by Cordeiro, therefore is going to extend the dataset for the detection task.

We also employed the YOLO network, which is the state-of-art in object detection task. The network was trained first to learn the general features for histopathological images, then it was retrained to learn features from HE images of each patient and tested on IHC images of that same patient. Our method achieved an average IoU score of 0.2094 at the standard deviation of 0.2522.

Finally, we conclude that the test sample detection on IHC slides is not a trivial task. It is necessary a combination of IHC color and texture information, besides shape features that were extracted from the HE slides. We can attribute the difficulty of this task due to the variety of sample shapes, color, texture, contours and staining quality.

6.1 FUTURE WORK

Improving the accuracy of a deep neural network is a complex task since there are many variables that can be adjusted which can affect our result. Below we cite some future work that can be done for improving the results of this approach.

- Classify tissue samples in further classes accordingly to their features;
- Train with larger datasets to increase the data volume and improve IoU rates. The dataset should have at least 2000 examples of each class varying in size, illumination, rotation, and scales;
- Train using patches of the input to decrease the resize done by the YOLO, therefore decreasing also the information loss and increasing the data;
- Use another neural network or other feature extractors to compose a more complex feature vector;
- Change hyper-parameters, e.g. try different values for learning rate.

REFERENCES

[1] Saad Ullah Akram, Juho Kannala, Lauri Eklund, and Janne Heikkilä. Cell proposal network for microscopy image analysis. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 3199–3203. IEEE, 2016.

[2] AlexeyAB. Yolo-v3 and yolo-v2 for windows and linux. `https://github.com/AlexeyAB/darknet`, 2019. Acessed on 06/10/2019.

[3] Raja S Alomari, Ron Allen, Bikash Sabata, and Vipin Chaudhary. Localization of tissues in high-resolution digital anatomic pathology images. In *Medical Imaging 2009: Computer-Aided Diagnosis*, volume 7260, page 726016. International Society for Optics and Photonics, 2009.

[4] Raja S Alomari, Subarna Ghosh, Vipin Chaudhary, and Omar Al-Kadi. Local binary patterns for stromal area removal in histology images. In *Medical Imaging 2012: Computer-Aided Diagnosis*, volume 8315, page 831524. International Society for Optics and Photonics, 2012.

[5] Teresa Araújo, Guilherme Aresta, Adrian Galdran, Pedro Costa, Ana Maria Mendonça, and Aurélio Campilho. Uolo-automatic object detection and segmentation in biomedical images. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, pages 165–173. Springer, 2018.

[6] Anup Bhande. What is underfitting and overfitting in machine learning and how to deal with it. `https://medium.com/greyatom/what-is-underfitting-and-overfitting-in-machine-learning-and-how-to-deal-with-it-6803a989c76`, 2018. Acessed on 06/06/2019.

[7] Freddie Bray, Jacques Ferlay, Isabelle Soerjomataram, Rebecca L Siegel, Lindsey A Torre, and Ahmedin Jemal. Global cancer statistics 2018: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: a cancer journal for clinicians*, 68(6):394–424, 2018.

[8] Caroline Quadros Cordeiro. An automatic patch-based approach for her-2 scoring in immunohistochemical breast cancer images. Master's thesis, Programa de Pós-Graduação em Informática - Universidade Federal do Paraná, Curitiba - PR, Setember 2018. Documento apresentado como requisito parcial para o exame de qualificação.

[9] Nvidia Developer. Convolution. `https://developer.nvidia.com/discover/convolution`, 2019. Acessed on 06/06/2019.

[10] Yuhang Dong, Hongda Shen, and W David Pan. An interactive tool for roi extraction and compression on whole slide images. In *2016 IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI)*, pages 224–227. IEEE, 2016.

[11] Rayachoti Eswaraiah and E Sreenivasa Reddy. Medical image watermarking technique for accurate tamper detection in roi and exact recovery of roi. *International journal of telemedicine and applications*, 2014:13, 2014.

[12] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.

[13] Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE transactions on pattern analysis and machine intelligence*, 32(9):1627–1645, 2009.

[14] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

[15] Enrique Dìaz Canton Gonzalo Jr Recondo, Màximo de la Vega, Martin Greco, and Matias E Valsecchi Gonzalo Sr Recondo. Therapeutic options for her-2 positive breast cancer: Perspectives and future directions. *World journal of clinical oncology*, 5(3):440, 2014.

[16] Manu Goyal and Moi Hoon Yap. Region of interest detection in dermoscopic images for natural data-augmentation. *arXiv preprint arXiv:1807.10711*, 2018.

[17] Simon Haykin. *Neural Networks and Learning Machines*. Pearson, 2009.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[19] Hazem Hiary, Raja S Alomari, and Vipin Chaudhary. Segmentation and localisation of whole slide images using unsupervised learning. *IET image processing*, 7(5):464–471, 2013.

[20] David G Hicks and Linda Schiffhauer. Standardized assessment of the her2 status in breast cancer by immunohistochemistry. *Laboratory Medicine*, 42(8):459–467, 2011.

[21] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[22] Zhe Li Caiwen Ding Bo Yuan Qinru Qiu Yanzhi Wang Ji Li, Ao Ren. Towards acceleration of deep convolutional neural networks using stochastic computing. `http://www.aspdac.com/aspdac2017/archive/pdf/2S-4_add_file.pdf`, 2018. Acessed on 06/08/2019.

[23] Tan Xiao Jian, Nazahah Mustafa, Mohd Yusoff Mashor, and journal=Journal of Engineering Research and Education volume=10 pages=57-72 year=2018 publisher=PENERBIT UNIMAP Ab Rahman, Khairul Shakir. Image processing in breast carcinoma histopathological image: A review.

[24] Pavel Kisilev, Eli Sason, Ella Barkan, and Sharbell Hashoul. Medical image captioning: learning to describe medical image findings using multi-task-loss cnn. 2011.

[25] Frederik Kratzert. Understanding the backward pass through batch normalization layer. `https://kratzert.github.io/2016/02/12/understanding-the-gradient-flow-through-the-batch-normalization-layer.html`, 2016. Accessed on 06/06/2019.

[26] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[27] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436, 2015.

[28] Yi Li. Detecting lesion bounding ellipses with gaussian proposal networks. *arXiv preprint arXiv:1902.09658*, 2019.

[29] Zhe Li, Chong Wang, Mei Han, Yuan Xue, Wei Wei, Li-Jia Li, and Li Fei-Fei. Thoracic disease identification and localization with limited supervision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8290–8299, 2018.

[30] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[31] Le Lu, Yefeng Zheng, Gustavo Carneiro, and Lin Yang. Deep learning and convolutional neural networks for medical image computing. *Advances in Computer Vision and Pattern Recognition; Springer: New York, NY, USA*, 2017.

[32] Michael A. Nielsen. *Neural Networks and Deep Learning*. Determination Press, 2015.

[33] World Health Organization. Who | cancer. `https://www.who.int/cancer/en/`, 2018. Acessed on 06/11/2019.

[34] Luis Perez and Jason Wang. The effectiveness of data augmentation in image classification using deep learning. *arXiv preprint arXiv:1712.04621*, 2017.

[35] Foster Provost and R Kohavi. Glossary of terms. *Journal of Machine Learning*, 30(2-3): 271–274, 1998.

[36] Prajit Ramachandran, Barret Zoph, and Quoc V Le. Searching for activation functions. *arXiv preprint arXiv:1710.05941*, 2017.

[37] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7263–7271, 2017.

[38] Joseph Redmon and Ali Farhadi. Yolo: Real-time object detection. `https://pjreddie.com/darknet/yolo/`, 2018. Acessed on 06/10/2019.

[39] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

[40] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[41] Stephanie Robertson, Hossein Azizpour, Kevin Smith, and Johan Hartman. Digital image analysis in breast pathology—from image processing techniques to artificial intelligence. *Translational Research*, 194:19–35, 2018.

[42] Adrian Rosebrock. Intersection over union (iou) for object detection. `https://www.pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/`, 2016. Acessed on 06/09/2019.

[43] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

[44] Jürgen Schmidhuber. Deep learning in neural networks: An overview. *Neural networks*, 61: 85–117, 2015.

[45] Towards Science. Demystifying convolutional neural networks. `https://towardsdatascience.com/demystifying-convolutional-neural-networks-384785791596`, 2018. Acessed on 06/06/2019.

[46] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[47] Tzutalin. Labelimg. `https://github.com/tzutalin/labelImg`, 2015. Acessed on 04/25/2019.

[48] Mitko Veta, Josien PW Pluim, Paul J Van Diest, and Max A Viergever. Breast cancer histopathology image analysis: A review. *IEEE Transactions on Biomedical Engineering*, 61(5):1400–1411, 2014.

[49] Athanasios Voulodimos, Nikolaos Doulamis, Anastasios Doulamis, and Eftychios Protopapadakis. Deep learning for computer vision: A brief review. *Computational intelligence and neuroscience*, 2018, 2018.

[50] JD Webster and RW Dunstan. Whole-slide imaging and automated image analysis: considerations and opportunities in the practice of pathology. *Veterinary pathology*, 51(1): 211–223, 2014.

[51] Dana Carmen Zaha. Significance of immunohistochemistry in breast cancer. *World journal of clinical oncology*, 5(3):382, 2014.

[52] Junkang Zhang, Haigen Hu, Shengyong Chen, Yujiao Huang, and Qiu Guan. Cancer cells detection in phase-contrast microscopy images based on faster r-cnn. In *2016 9th International Symposium on Computational Intelligence and Design (ISCID)*, volume 1, pages 363–367. IEEE, 2016.