Polyp Detection using Convolutional Neural Networks: An Exploratory Study

Javier Rodríguez Carmona

Resumen—. Este proyecto presenta un estudio exploratorio sobre el potencial de usar redes neuronales convolutivas (CNNs) para ayudar en tareas de detección de pólipos. La detección temprana de pólipos es crucial para la supervivencia del paciente, y podría ser muy útil para los médicos disponer de una herramienta que les ayude a identificar pólipos en tiempo real en las imágenes. El principal reto al que nos enfrentamos es trabajar en un dominio en el que los sets de datos disponibles públicamente son escasos y no están balanceados. Nuestro objetivo es comprobar el potencial de usar redes preentrenadas y fine-tuning para aprovechar las capacidades de las CNNs. Proponemos dos estrategias diferentes: clasificación usando las imágenes enteras y clasificación basada en trozos de imágenes o patches. Como las bases de datos que usamos para validar nuestra metodología no están balanceadas, también exploraremos el uso de técnicas de early data augmentation para incrementar el tamaño del dataset. Resultados preliminares indican el potencial de nuestra propuesta y comprueban cuantitativaente el impacto de cada una de las modificaciones que proponemos para el baseline. Más específicamente, el uso de clasificación basada en patches supera a la basada en imágenes, y hay un impacto demostrado en la eliminación del borde de las imágenes y en la aplicación de data augmentation. Finalmente, nuestros resultados muestran la importancia de tener datasets balanceados tanto en el entrenamiento como en el test de la red.

Palabras clave —Deep learning, redes neuronales convolutivas, detección de pólipos, data augmentation, imagen médica, machine learning.

Summary—. This project presents an exploratory study on the potential of using convolutional neural networks (CNNs) to aid in polyp detection tasks. Early polyp detection is crucial for patient survival and it could be very useful for clinicians to have a tool that could aid clinicians in real time to identify polyps in the image. The main challenge we faced was to work in a domain where publicly available datasets are scarce and, moreover, they are not balanced, we aim to assess the potential of using pre-trained networks and fine-tuning to take profit of the capabilities of CNNs. We propose two different strategies: classification using the whole image and patch-based classification. As the databases that we use to validate our methodology are clearly not balanced, we also explore the use of early image augmentation techniques to increase dataset size. Preliminary results indicate the potential of our proposal and assess quantitatively the impact of each of the modifications we propose to the baseline. More precisely, the use of patch-based classification outperforms image-based one, and there is a proven impact of supressing image borders and performing data augmentation. Finally, our results shows the importance of having balanced datasets for both training and testing of the network.

Index Terms —Deep learning, convolutional neural networks, polyp detection, data augmentation, medical image, machine learning.

1 Introduction

Colorectal cancer is nowadays the fourth cause of cancer death worldwide; for instance, 95,270 new cases are expected in USA during 2016 as reported by American Cancer Society [1]. Early detection of colorectal cancer precursor lesion, polyps, is crucial for patient survival in a way such the earlier they are detected, the more likely the patient to survive.

Several colon screening methods are used for polyp detection although colonoscopy is still considered as the gold standard. Nevertheless, colonoscopy presents some drawbacks being the most important of them polyp miss-rate [2], that is, some polyps are missed during the procedure. Apart from this, current trends indicate to remove the polyp when it is found, regardless of its status (benign or malign). Once polyps are removed, they are all sent to histology analysis which, in the case of benign polyps, would suppose a waste of resources. Moreover, having to wait until histology results delays the start of patient treatment.

Taking all this into account, there is room for developing computational tools that can aid clinician in polyp detection and, once polyps are detected, in polyp classification by means of the analysis of the content of the detected polyp regions.

In this project, we tackle the first task – polyp detection – and we propose the use of CNNs as a first feasible solution for a potential application in the intervention room. Having an automatic system that can solve detection problems would suppose a breakthrough for colorectal cancer detection which might suppose a decrease in medical costs to the healthcare system and, more importantly, to increase patients' survival rate.

- *E-mail de contacte: javierrc13@gmail.com*
- *Menció realitzada: Computació*
- *Treball tutoritzat per: Jorge Bernal y David Vázquez (Ciències de la Computació)*
- *Curs 2015/16*

1.1 Motivation

The main motivation of this project is to apply the technologies and tools learned during the Computer Science degree in order to solve real life problems in challenging domains such as, in this case, polyp detection. Out of the tools that were presented during machine learning courses, we think it would be really interesting to extend the knowledge acquired on CNNs, as their popularity and use has suffered a huge boost in the later years.

1.2 Objectives to achieve

The main objective of this project is to assess the potential of convolutional neural networks to build up reliable polyp detection systems. In order to achieve this goal, we propose to tackle the following sub-objectives:

- Extensive study over a pre-trained network in order to learn what each layer does and how different layers can be combined to improve global performance.
- Explore and compare different detection approaches: image-based or patch-based.
- Study on different alternatives related to how input data should be fed into the network to solve some of the problems inherent to the domain of application (small and not balanced databases).
- Proposal of a robust validation framework including the specification of the validation database and the definition of adequate performance metrics.
- Perform extensive tests to assess the impact of each stage of our proposed methodology.

1.3 Project methodology

To develop this project, the chosen methodology was SUM. This choice was done considering the type of project to be carried out – one-man team-; this methodology allows to achieve maximum efficiency using iterative weekly sprints on which the work done the last week is analysed and new tasks are defined for the next one.

1.4 Project requirements

1.4.2 Hardware requirements

The compulsory requirement to carry out this project was to have access to a PC with a GPU able to run CUDA and CUDNN so training and testing of networks can be performed in a reasonable amount of time. We present minimum and tested requirements in Table 1.

Table 1.Hardware requirements.

	Minimal	**Tested**
CPU	Intel Core i5	Intel Core i7-6700
RAM	4GB	16GB
GPU	NVidia GeForce 410M	NVidia GeForce GTX 970

1.4.3 Software requirements

We present in Table 2 minimal and tested software requirements to develop this project:

Table 2. Software requirements.

	Minimal	**Tested**
OS	Ubuntu 14.04 LTS	Ubuntu 14.04 LTS
CUDA	5.5	6.5
Matlab	R2013b	R2014b
Matconvnet	Matconvnet V1.0-beta0	Matconvnet V1.0-beta18

1.5 Project planning

In order to achieve the objectives of the project, we propose the following tasks:

- Bibliographic research.
- Studying the networks available in the literature and choosing the most appropriate one.
- Performing the first validation experiments.
- Developing pre-processing techniques and reconfiguring the network as for example switching between patch-based and image-based classification.
- Final validation experiments.

We show corresponding Gantt diagram in the Annex.

1.6 Document structure

After this introduction, we will introduce the state-of-the-art on neural networks. Next, we will explain our complete methodology for polyp detection using CNNs, which includes the description of the network used and the definition of pre-processing operations applied to input images. After this, we present in Section 4 the complete validation framework, including the definition of validation databases and metrics along with presenting the experiments to be carried out. In Section 5 we present experimental results along with an in-depth analysis of the results obtained. Finally, we close the paper in Section 6 presenting the main conclusions extracted along with introducing future work planned to be done.

2 STATE OF THE ART

2.1 Introduction

Historically, several algorithms and methods have been used in order to apply machine learning, amongst them there are *naïve Bayes [3]*, specialized on learning from low amounts of data, although it doesn't seem to learn well the relation between different features. Another one is *logistic regression [4]*, which can be easily adapted to collect new data to update a previous model. *Decision trees* [5] are also a type of machine learning, which present problems related to having to rebuild the tree once new examples come

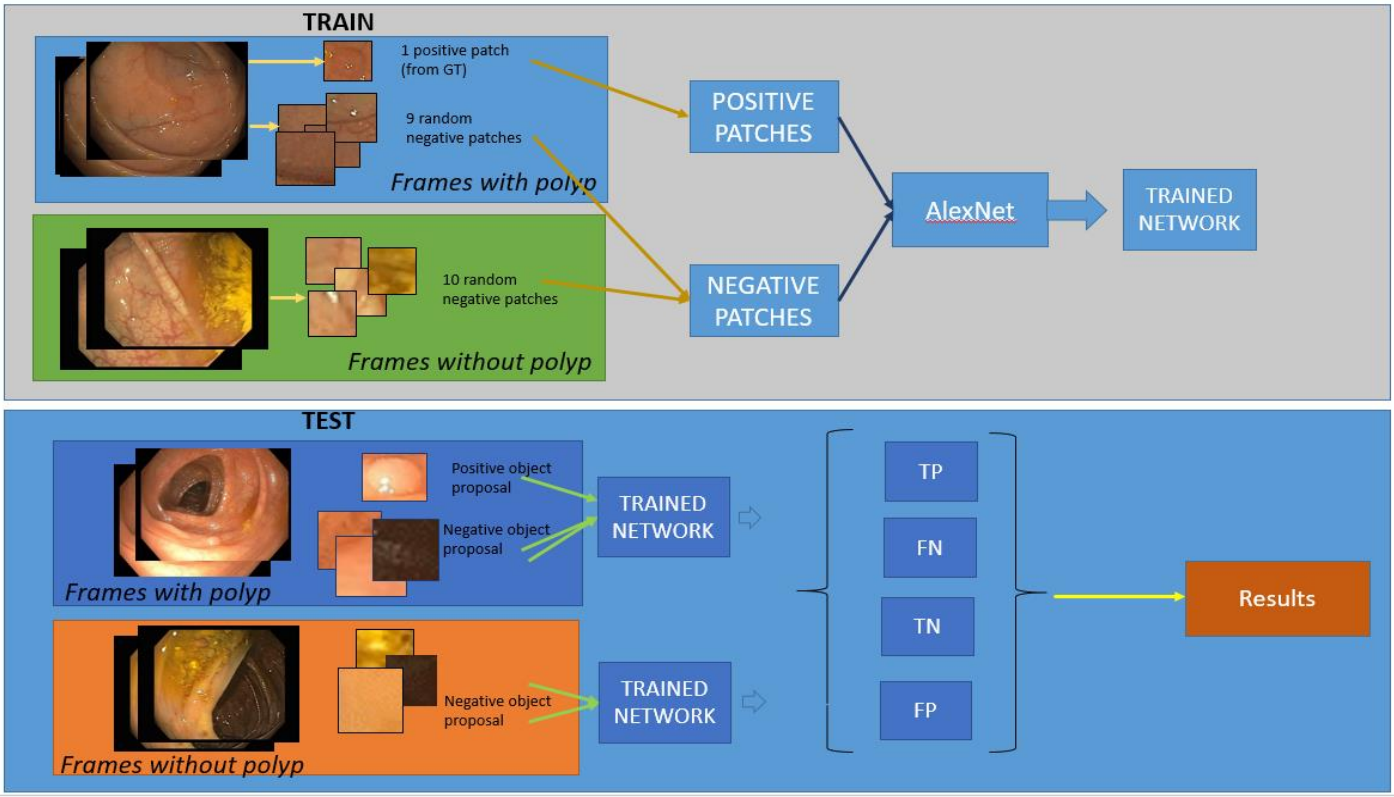


Figure 1. Patch-based polyp detection processing scheme

in and tending to overfit their results. Furthermore, *SVMs [6]* would be another example with proven good performance though they tend to consume a lot of memory and are hard to tune. Finally, *neural networks* usually present issues at the training stage by requiring huge amounts of data and taking long periods of time in order to finish training. However, recent use of fine tuning and CUDA [7] help reducing both the data and time requirements.

2.2 Neural networks

Artificial Neural Networks (ANN) [8] are a group of models based on biological neural networks. ANNs are formed by several sets of hidden layers connected between them by weighted connections. On the training stages, 2 events are triggered, forward-propagation and backward propagation.

On forward-propagation, inputs run through the layers activating certain neurons and their respective connecting weights in order to obtain an output at the end of the network. On backward propagation, accordingly on the accuracy of the previously obtained results, the weights are updated from the output to the input, thus allowing the network learn the necessary data. On the testing phase, only the forward propagation is executed, as the weights no longer need to be updated.

There are different kinds of neural networks, as for example convolutional neural networks – CNNs - [9], which are the ones to be used in this project. Also, fine-tuning [10] can be done to a network to take advantage of the generic data learnt from another already trained network and decrease the amount of data and time required to train a network from scratch.

CNNs have their origins on visual learning done by alive organisms which was discovered by 1968, this knowledge later evolved into neocognitrons [11]. Neocognitrons are a kind of hierarchical artificial neural networks proposed by Kunihiko Fukushima in the 1980s that were specifically used for handwritten character recognition and other tasks on recognition of patterns. The main difference between the neocognitron and convolutional neural networks is that the later forces the same trainable weights at different points of the network. This idea was originated in 1986, the year when CNNs were 'born' though, as it is clear, their design has been iterated and improved, but their concept has perdured till today.

These kinds of networks have been used for different purposes, from image classification with databases such as MINST database [12], to video analysis [13], to something as different as playing go in December 2014 on an experiment conducted by Christopher Clark and Amos Storkey, obtaining some success on doing so [14]. This is just a glimpe of the wide variety of uses on the current literature.

Though CNNs have been applied in a variety of domains, their use in medical image analysis is still in its early stages. The reason behind this is the lack of publicly available annotated databases which could be used for building networks from scratch. Apart from this, in cases where they are available, we do not always have a same number

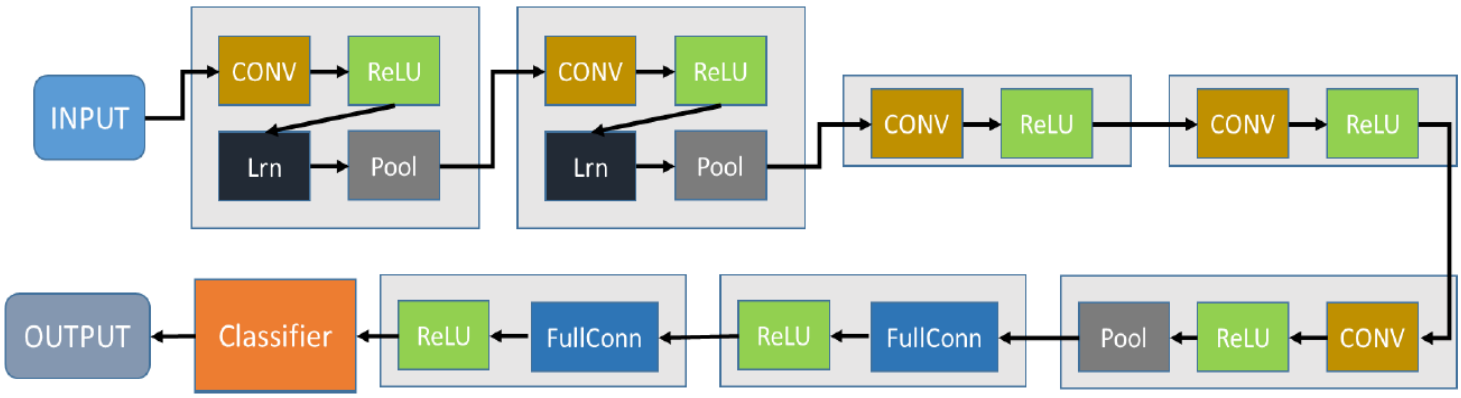


Figure 2. Network definition

of quality positive and negative examples, which may result on introducing undesirable bias to the network. We explore in this project how a widespread-used pre-trained network could be used as part of a medical image classification system. More precisely, we aim to explore their use in polyp detection tasks, in which the system should determine polyp presence/absence by the analysis of either the whole image or small patches extracted from it.

3 METHODOLOGY

3.1 INTRODUCTION

Before presenting the whole methodology, it is important to mention that within this work we will test both image-based and patch-based polyp detection. The differences between them are the following: in the former, the network will be trained and tested with full images, and the classifier layer will determine whether there is a polyp in the image or not. In patch-based classification, we feed the network with patches containing examples of polyp and non-polyp regions in the image. In the same way, to produce the final output of our method, the system will classify patches and polyp presence/absence will be determined by combining information of all the patches extracted from the testing image –see Figure 1-.

The complete classification methodology works in the following way: images/patches are created from input data and those which are part of the training set are run through the network both backwards and forward in order to perform network training. Input data is analysed for as many epochs (iterations) as needed until the network converges. Once the network is trained, images/patches from the test set are fed to the network in order to obtain final detection score.

3.2 Pre-trained network description

We propose to use Alex-net as our pre-trained network. Alex-net has the following network configuration:

- Two consecutive sets containing the following sequence of layers: convolution-ReLU-learning-pooling.
- Two consecutive sets containing the following sequence of layers: convolution-ReLU.
- A set containing the following sequence of layers: convolution-ReLU-pooling.
- Two sets containing the following sequence of layers: fully connected-ReLU
- Final classifier layer.

We show a scheme of the network in the Figure 2. The Alex-net network has been previously used specifically for object detection and image detection on a wide scope of objects in order to detect different object in an image and their position relative to each other. For example, it was originally used for the ImageNet challenge [15].

Alex-Net network contains the following layers [16], which also appear in other CNNs configurations:

- Convolutional layer.
  A convolutional layer provides the ability to create an output stimuli from an input one, which also includes storing the information associated to the weights of the neurons involved. It is a 3-D layer that generates a set of filters from a receptive field. These filters use local connectivity, which is especially useful while learning from high dimensional images but avoiding the use of full connectivity, which would heavily affect the amount of GPU memory used.
  This local connectivity can be customized in several ways: *depth* controls the amount of neurons that will connect to the same input region, *stride* controls the width and height allocation for the input and finally *padding* consists of a filling of zeros around the input –padding can be used for optimization purposes -.
- Pooling layer.
  A pooling layer implements a non-linear down-sampling which segments the image in regions that do not overlap and generates an output with the maximum value from each region. This layer should not suppose a relevant loss of information, as the goal is to obtain its most representative feature and the general location of the region in relation with the others. Its most straightforward benefit is a decrease in the spatial size of the network and that it allows to control overfitting

by decreasing the amount of parameters.

- ReLU (Rectified Linear Units) layer.
  This layer is used in order to de-linearize the decision function of the neural network in order to provide a different kind of neuron response to a stimuli. De-linearization is done by applying a non-saturating activation function.
- SoftMax (Rectified Linear Units) layer.
  A SoftMax layer is usually used as a final layer on classification networks, it converts the output from the network into probabilities, ranging from 0 to 1. This conversion allows an easier interpretation of the result and gives data about the confidence of that result.

3.3 Input data preparation

Considering the domain of application and some particularities that colonoscopy images present, we propose in this section several image pre-processing strategies in order to provide the network with best possible input data to help for classification purposes.

3.3.1 Border cutting

As colonoscopy images present an inherent black mask to surround the scene – to avoid showing areas with low quality due to how images are acquired – it is important to deal with these black borders as, in some cases, they can take about 20% of the image. Removing information associated to them is crucial in order to avoid the network learning something outside our target structure, helping the network to focus on learning polyps and intestine walls appearances instead of introducing noisy information associated to them. It is important to mention that border removal has been done automatically and considers unexpected variability of border size even within a video.

3.3.2 Early data augmentation

a) b) c) d)

Figure 3. Early data augmentation (a is the original, b is the flip on both axis, c is the flip on the x axis and d is the flip on the y axis)

As there was a huge imbalance between positive and negative examples, we decided to employ an early type of data augmentation in order to artificially boost the amount of samples. Said data augmentation consisted on applying horizontal, vertical and both flips to the images, therefore multiplying by 4 the amount of samples on the dataset. The results can be seen on Figure 3.

3.3.3 BGR images

As we were not achieving the best results, we started investigating and we came across some sources that stated that AlexNet was originally trained with the images set up in BGR channels instead of RGB, so we decided to check whether change in color channel ordering has any impact on final classification results or not.

3.3.4 Patch extraction from images (only for patch-based polyp detection)

In order to test different alternatives for polyp detection using CNNs, we studied the possibility of using image patches instead of using the full image, aiming to train the network with more specific polyp information, as a bounding box around the polyp will represent better a polyp than the whole image).

We propose the following strategy for patch extraction:
a) Positive patches: in every frame with a polyp we extract a sub-image (patch) which covers the whole polyp.
b) Negative patches: we studied two different options regarding non-polyp patches:

- The first one consisted on dividing the image in 16 parts (4 horizontal and 4 vertical divisions). Out of these 16 patches, we kept those which did not contain polyp information and we stored for later use only those which presented very low standard deviation of their greyscale value, under the assumption that this would represent a very flat texture which could be typical of the mucosa wall.
- The second one eliminates the rigidness of dividing an image on equal-size squares and bases patch creation in choosing random points in the image and, from them, selecting an image patch centered on them, taking into account that no polyp information should be present on them.
- The first strategy may lead to faster implementation at the cost of providing too-uniform regions that can be very similar between images (as there is mucosa in all images. The second one would provide with more variety in terms of negative patch appearance at the cost of larger processing time. We will test the impact of negative-patch selection in Section 5.

3.4 Calculation of classifier output

3.4.1 Image-based classifier

For this specific classifier, each frame is run through the network. The classifier would assign a likelihood/score of the image to belong to any of the two classes (polyp and no-polyp) and the system takes as final decision the one with higher score.

3.4.1 Patch-based classifier

Regarding patch-based classification, two different strategies were studied: the first one consists of feeding the network with positive and negative patches from each image in the training set, validating the network patch by patch (not providing final global image classification) whereas the second one does use patch information from a same image to reach a final decision for the global image.

Table 3. Database description.

CVC-CLINIC		AsuMayo			
Train		Train		Test	
Positive	Negative	Positive	Negative	Positive	Negative
37.074 (73%)	13.500 (27%)	5.402 (34.7%)	10.121 (65.3%)	3.975 (23.22%)	13.147 (76.78%)

With respect to the second strategy, each of the test image patches are run through the network obtaining a polyp likelihood score for each one. In this case, we propose to select the patch with higher score as the one more likely to contain a polyp in the image. As not all images contain a polyp, we will also run a study on this specific score achieved by the candidate patch in order to assess whether there is a threshold value that could be used to discard images without polyp.

4. EXPERIMENTAL SETUP

4.1 DATABASE DESCRIPTION

We validate our methodology in the following 3 databases ,whose proportions can be seen on Table4.

- CVC-CLINIC [17]: This database is formed by 37.074 polyp, 13.500 non-polyp frames with a pretty stable resolution amongst them as can be seen in Figure 4.

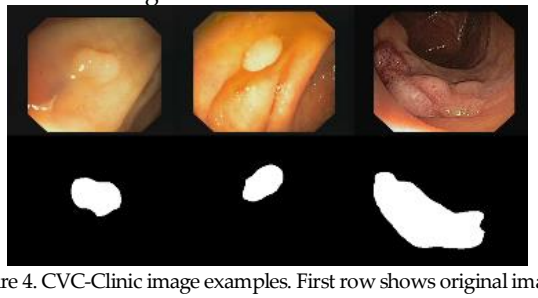


Figure 4. CVC-Clinic image examples. First row shows original images whereas second row shows binary masks representing actual polyp location in the image

- ASU-MAYO train [18]:

This database is formed by 5.402 polyp, 10.121 non-polyp frames with several different resolutions between them.

- ASU-MAYO test [19]:

This database is formed by 3.975 polyp, 13.147 non-polyp frames with really different resolutions between them as can be seen in Figure 5.

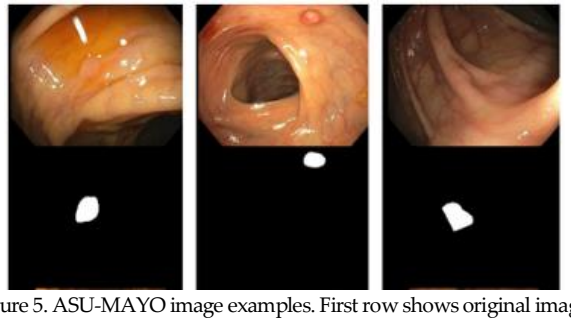


Figure 5. ASU-MAYO image examples. First row shows original images whereas second row shows binary masks representing actual polyp location in the image

4.2 PERFORMANCE METRICS

By comparing the output provided for each image/patch classified with associated ground truth, we can calculate the following 4 values [20]: TP (True Positive), FP (False Positive), TN (True Negative) and FN (False Negative). These are obtained by following this set of rules in Table 4:

Table 4. Metrics definition.

Our system/Ground truth	Polyp in the image	No polyp in the image
Predicts polyp presence	TP	FP
Does not predict polyp presence	FN	TN

Using these definitions we can calculate the following performance metrics

Precision=TP/(TP+FP)
Recall= TP/(TP+FN)
Specificity=TN/(FP+TN)
Accuracy=(TP+TN)/(TP+TN+FP+FN)
F1 score = 2*((precision*recall) /(precision+recall))
F2 score= 5*((precision*recall) /(4*precision+recall))

It is worth to note that, considering the specific application we are working at, it is highly important to use measures such as F2-score which punish false negative values which would represent the loss of a polyp. This loss would potentially put at risk the health of the patient, while a false positive value would only make the clinician waste time in the exploration which would have an impact in the efficiency of the public healthcare system.

4.3 PROPOSED EXPERIMENTS

4.3.1 IMAGE-BASED CLASSIFICATION

Firstly, we propose to make some experiments on training and testing with full images; the procedure consists of the following steps:

- Image pre-processing (black border cutting, image resize to fit expected size by the network)
- Train the network using a training database until it converges and validate the trained network over a different dataset.

In this experiments we test the use of several combinations of available datasets and we even consider the use of a different pre-trained network. We aim to assess the importance of databases used in detection results and the impact of some pre-processing operations.

Table 5. Image classification results

Config	Train-set	Test-set	Pre-processing	Prec	Rec	F1	F2
2	CVC-Clinic	Asu-Mayo Test	None	33,11%	87,38%	48,03%	65,81%
3	Asu-Mayo Train	Asu-Mayo Test	None	99,70%	48,98%	65,69%	54,53%
4	Asu-Mayo Train	CVC-Clinic	None	88,02%	13,08%	22,77%	15,76%
5	CVC-Clinic	Asu-Mayo Test	Border removal	18,30%	39,54%	25,02%	32,09%
6	Asu-Mayo Train	Asu-Mayo Test	Border removal	28,71%	99,98%	44,61%	66,81%
7	Asu-Mayo Train	CVC-Clinic	Border removal	78,61%	99,78%	87,93%	94,68%

Table 6. Patch classification results

Config	Train-set	Test-set	Pre-processing	Prec	Rec	F1	F2
14	Asu-Mayo Train	Asu-Mayo Test	Border removal, AlexNet	99,59%	18,49%	31,19%	22,09%
16	Asu-Mayo Train	Asu-Mayo Test	Border removal, AlexNet, removal of half negative examples	50,29%	78,19%	61,21%	70,38%
18	Asu-Mayo Train	CVC-Clinic	Border removal, CVC-clinic is full images, not patches	78,61%	98,09%	87,27%	93,46%

4.3.2 PATCH-BASED CLASSIFICATION

Patch-based detection experiments involve some of the aspects from the previous experiments and adds new items to be evaluated such as the impact of image augmentation and the degree of balance achieved after pre-processing operations.

Patch-based detection is validated in two different ways: in the first one we validate patch classification individually, independent of whether they are extracted from the same image or not. That is, we have a classification score for each patch but we do not take decisions of polyp presence in a given image as we do not integrate information from all patches extracted for this particular image. In the second experiment, we do integrate individual patch classification score to reach a decision about polyp presence or absence in the image: we take the patch with maximum confidence value as object proposal and we study whether this confidence value can be used as a threshold to determine polyp presence.

5. EXPERIMENTAL RESULTS

5.1 IMAGE CLASSIFICATION

5.1.1 DATABASE IMPACT

As we can see on the results on Table 5, both databases tend to overfit - CVC-Clinic to the positives, and Asu-Mayo to the negatives – as the system obtains especially bad results when both databases are used within a same validation experiment. Results make more evident this overfitting as in for example in the configuration 4 where the network achieved a high precision, but a low recall, showing that one of the classes is being predicted really well, but the other one isn't.

5.1.2 IMPACT OF INPUT DATA

From what we can see, removing the black margins from the Asu-Mayo frames caused an impact, but with mixed results, as for example, the configuration 5 got worse scores than the 2, but the configurations 6 and specially 7 did significantly better than the 3 and 4. We associate this to being configuration 5 trained with CVC-Clinic and it still has some black borders which it might learn and fails to detect on the borderless asu-mayo image

5.1.3 INITIAL CONCLUSIONS

From what we can observe, training the network after removing the black borders seemed to improve greatly the results obtained from it and that we should stick to one of the databases for validation as using both of them in the same experiment may lead to overfitting. Furthermore, as the databases are not balanced, we are experiencing the issue where the network learns only one of the sets and ignores the other one.

Table 7. Classic patch classification results

Config	Train-set	Test-set	Pre-processing	Prec	Rec	F1	F2
19	Asu-Mayo Train	Asu-Mayo Test	Removal of half negative examples, RGB image	50,29%	78,19%	61,21%	70,38%
20	Asu-Mayo Train	Asu-Mayo Test	Removal of half negative examples, BGR image	51,30%	87,60%	64,71%	76,74%
21	Asu-Mayo Train	Asu-Mayo Test	One negative example per positive example, RGB image	52,29%	78,24%	62,68%	71,17%
22	Asu-Mayo Train	Asu-Mayo Test	One negative example per positive example, BGR image	67,23%	78,42%	72,40%	75,89%
23	Asu-Mayo Train	Asu-Mayo Test	Same amount of positives and negatives, negatives ordered by std deviation, RGB image	5,94%	96,88%	11,20%	23,85%
24	Asu-Mayo Train	Asu-Mayo Test	Same amount of positives and negatives, negatives ordered by std deviation, BGR image	10,60%	96,68%	19,11%	36,85%
25	Asu-Mayo Train	Asu-Mayo Test	19 with data augmentation	15,03%	88,38%	25,69%	44,72%
26	Asu-Mayo Train	Asu-Mayo Test	20 with data augmentation	5,40%	85,99%	10,16%	21,57%
27	Asu-Mayo Train	Asu-Mayo Test	21 with data augmentation	14,88%	66,99%	24,35%	39,39%
28	Asu-Mayo Train	Asu-Mayo Test	22 with data augmentation	31,26%	88,81%	46,24%	64,91%
29	Asu-Mayo Train	Asu-Mayo Test	23 with data augmentation	10,63%	96,23%	19,15%	36,86%
30	Asu-Mayo Train	Asu-Mayo Test	24 with data augmentation	27,64%	93,74%	42,69%	63,41%

5.2 PATCH CLASSIFICATION

In this case we can observe on Table 6 that both configurations achieve high precision scores though the use of patches results in a higher number of true negatives. We have to consider here that global numbers are not comparable, as we do not have the same number of negative examples in both experiments – the network seems to be classifying each patch as negative- .

5.2.1 CLASSIC PATCH CLASSIFICATION

On this specific experiment, we performed a series of exhaustive experiments on which we tested how each configuration variable affected the result of the network training. The results can be observed on table 7.

- Impact of BGR

The main conclusion that we can extract is that interchanging input channels has a positive impact in performance scores. In this case, we can also observe that extracting negative patches in a global way (after analysing all the images) has a negative impact in the results, especially in terms of false positives.

- Impact of data augmentation

Regarding the impact of data augmentation, we can observe that it does not lead to an improvement bar some particular cases. We cannot observe any trend regarding its impact, as for some metrics it helps but for other it provides worse results. Surprisingly, it seems to improve more in cases where the use of BGR did not offer its best improvement – global selection of negative patches – therefore their expected combined improvement does not happen (which would be configuration 28).

- Database balance

The main conclusion that we extract from this experiment is that balancing does affect performance score. As positive and negative examples are more balance, performance gets better though in this case we find significant differences regarding the way we choose false negative patches, being the solution consisting of extracting them only from polyp frames the one with better performance.

- Impact of patch selection strategy

From the results obtained from this experiment, we can conclude that, getting the most uniform image by sorting them with their standard deviation improves learning what is negative. But at the cost of learning what is positive, as it can be seen on configuration 24, which compared to configuration 22 -that only lacks sorting by standard deviation- predicts around 56% less positives correctly.

5.2.2 IMAGE CLASSIFICATION BY PATCH

Our experiments over a fine-tuned AlexNet using images at a higher resolution did not really provide good results;

Table 8. Image classification by patch results

Config	DB used	Threshold	Pre-processing	Prec	Rec	F1	F2
33	Asu-Mayo	0	Further fine-tuned AlexNet, used 72x72 images. Applied data augmentation to training	0,15%	100,00%	0,29%	0,73%
		0.25		100,00%	40,00%	57,14%	45,45%
		0.5		100,00%	41,67%	58,82%	47,17%
		0.75		100,00%	29,17%	45,16%	33,98%

even after trying to resize our images to original patch size of the network -72x72-, we could not achieve results close close to ones obtained by using classic patch classification. The resulting network classified almost always everything as a no polyp. This could be easily explained, as in these experiments the training set had a higher amount of negative (180.404) examples than of positive ones (15.424), so it's understandable that the network decided to learn the negatives first.

- Impact of image resize

We tried training on a higher resolution than the one we tested on the rest of experiments, increasing from 72x72px to 227x227 px, the result of this experiment did not allow us to obtain any remarkable results or even near as good to the previous tested resolution, so it was determined that is not a good parameter to change.

5.3 Summary of results and proposed strategy

From all the results analysed previously, the best configuration to test in order to obtain good results would be one similar to configuration 22. This configuration included: patch-based classification, database balancing, feeding input data on BGR format, and using the training and testing ASU-MAYO databases as training and validation databases.

6 CONCLUSIONS

6.1 Conclusions from the project

The first conclusions would be that we have learnt that, though neural networks can indeed be a great tool for solving problems, their straightforward use may not be sufficient for applications like the one we propose here (general databases used in pre-trained networks contain human-made objects, polyps are not). Nevertheless, CNNs are an extremely potent model type and increasing knowledge about them would definitely lead us to improve current results in the future.

Furthermore, after testing both the image-based and the patch based approach, we can conclude that a patch-based approach offer better results that the image based one. This could be explained by the polyps being too small on the whole images and being ignored by the neural network, but when extracting patches, those polyps cover almost the whole image, allowing the network to learn more polyp-specific information. Also, if data augmentation is performed on the positive dataset, we can achieve a higher balance between positives and negatives, what proved to obtain better results.

In addition, the tests we performed on the pre-processing of the images obtained mixed results: the positive conclusions that can be extracted is that image and database preprocessing (removing the black borders from the images, feeding the images on BGR format and using the same amount of positive and negative samples) does indeed lead to an improvement of the results obtained from the network. However, other tested operations such as increasing the resolution of the images to be fed to the network and applying data augmentation to the positive dataset, which is the smallest one, caused the network to get worse results than what would be obtained otherwise.

Finally, the comparison between the two patch-based sub-strategies (classical patch classification and image classification with patches) led the first one to obtain better results as the second one failed to provide results close to the first one, so we conclude that for this particular problem, classic patch classification would be better.

6.2 Planification revision

During the project we encountered some issues that caused us to change the plannification. The first one was the initial setup of Matconvnet and the rest of the training libraries, which set us back a few weeks. The specific problem was getting the CUDA libraries to work with Matconvnet, ant we couldn't avoid using them as without CUDA the training times were excessively long. Also, the fact that the databases weren't balanced made it more difficult to go directly into training networks, this lead us to change our focus to working on the pre-processing of the data.

6.2 Future lines

Future research lines may consist on:

- Study of the potential of using available information to build a network from scratch
- Implement more advanced image pre-processing operations and study alternatives used in the literature when data is not balanced
- Test other pre-trained networks using different image databases.

Special thanks

Special thanks for my advisors, Jorge and David, for all the help and guiding that they provided me during the whole project. I would also like to thank my family too for the support they gave me during it too. Finally, thanks to Marc Masana for all the support he gave us with setting up the software and solving doubts.

BIBLIOGRAPHY

[1] The American cancer society. Key statistics for colorectal cancer.[http://www.cancer.org/cancer/colonandrectum-cancer/detailedguide/colorectal-cancer-key-statistics]

[2] Añade referencia a A. Leufkens, M. van Oijen, F. Vleggaar, and P. Siersema, "Factors influencing the miss rate of polyps in a back-to-back colonoscopy study", Endoscopy, vol. 44, no. 05, pp. 470–475, 2012

[3] Rish, I. (2001, August). An empirical study of the naive Bayes classifier. InIJCAI 2001 workshop on empirical methods in artificial intelligence (Vol. 3, No. 22, pp. 41-46). IBM New York.

[4] Hosmer, D. W., & Lemeshow, S. (2000). Introduction to the logistic regression model. Applied Logistic Regression, Second Edition, 1-30.

[5] Swain, P. H., & Hauska, H. (1977). The decision tree classifier: Design and potential. IEEE Transactions on Geoscience Electronics, 15(3), 142-147.

[6] Duan, K., Keerthi, S. S., & Poo, A. N. (2003). Evaluation of simple performance measures for tuning SVM hyperparameters. Neurocomputing, 51, 41-59.

[7] Nickolls, J., Buck, I., Garland, M., & Skadron, K. (2008). Scalable parallel programming with CUDA. Queue, 6(2), 40-53.

[8] Hagan, M. T., Demuth, H. B., Beale, M. H., & De Jesús, O. (1996). Neural network design (Vol. 20). Boston: PWS publishing company.

[9] Vedaldi, A., & Lenc, K. (2015, October). MatConvNet: Convolutional neural networks for matlab. In Proceedings of the 23rd Annual ACM Conference on Multimedia Conference (pp. 689-692). ACM.

[10] Kim, Y. (2014). Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882.

[11] Fukushima, K. (1988). Neocognitron: A hierarchical neural network capable of visual pattern recognition. Neural networks, 1(2), 119-130.

[12] Ciregan, D., Meier, U., & Schmidhuber, J. (2012, June). Multi-column deep neural networks for image classification. In Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on (pp. 3642-3649). IEEE.

[13] Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., & Fei-Fei, L. (2014). Large-scale video classification with convolutional neural networks. InProceedings of the IEEE conference on Computer Vision and Pattern Recognition (pp. 1725-1732).

[14] Clark, C., & Storkey, A. (2014). Teaching deep convolutional neural networks to play go. arXiv preprint arXiv:1412.3409.

[15] Olga Russakovsky*, Jia Deng*, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg and Li Fei-Fei. (* = equal contribution) ImageNet Large Scale Visual Recognition Challenge. IJCV, 2015.

[16] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems (pp. 1097-1105).

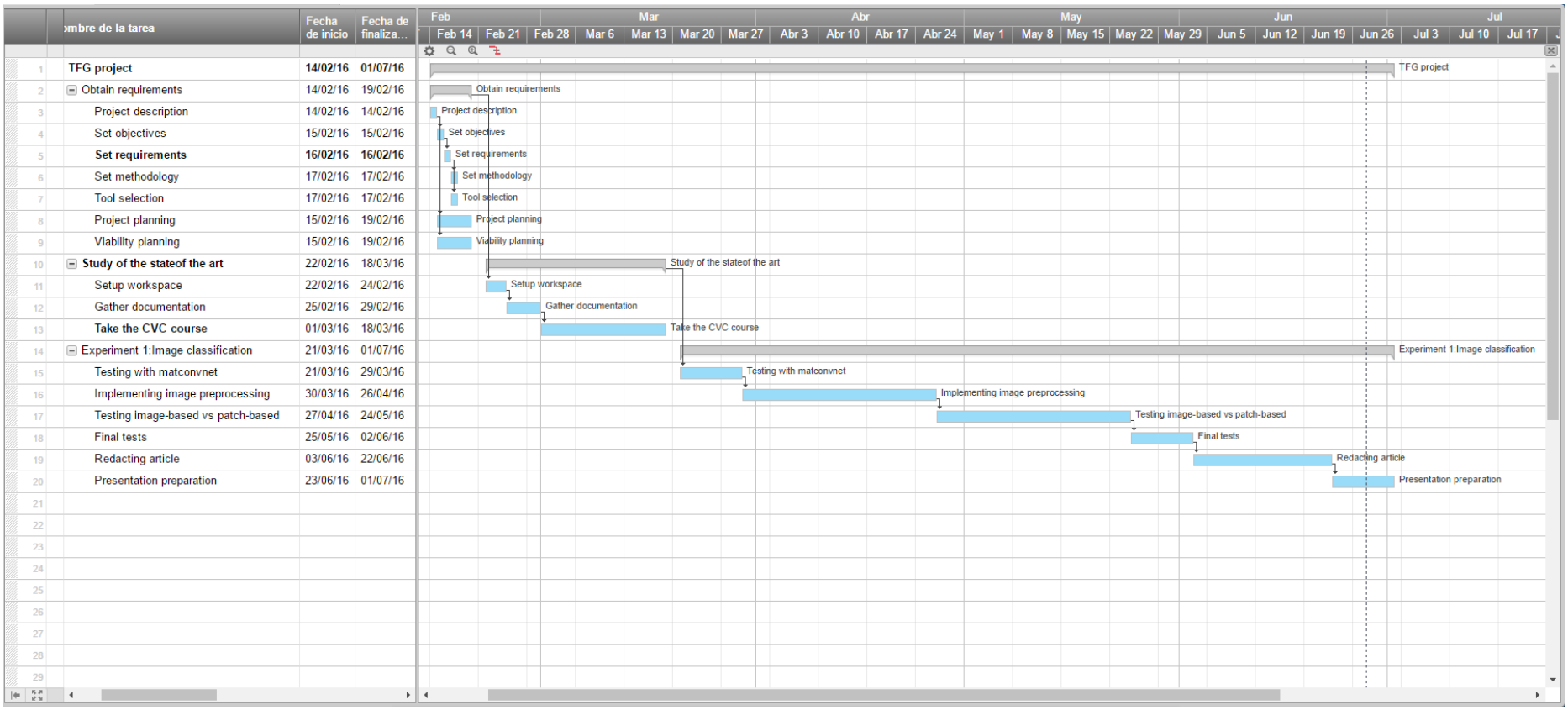
[17] "AsuMayo and CVC databases available to download(images and GT)." [Online]. Available: [http://polyp.grand-challenge.org/databases/]

[18] Bernal, J., Sánchez, F. J., Fernández-Esparrach, G., Gil, D., Rodríguez, C., & Vilariño, F. (2015). WM-DOVA maps for accurate polyp highlighting in colonoscopy: Validation vs. saliency maps from physicians. Computerized Medical Imaging and Graphics, 43, 99-111.

[19] Bernal, J., Tajkbaksh, N., Sánchez, F.J., Liang, J., Chen H., Yu, L., Angermann, Q., Romain, O., Rustad, B., Balasingham, I., Pogorelov, K., Riegler, M.A., Choi, S., Matuszewski, B., Debard, Q., Maier-Hein, L., Speidel, S., Stoyanov, D., Cordova, H., Sánchez-Montes, C., Gurudu, S.R., Fernández-Esparrach, G., Dray, X. and Histace, A. "Comparative Validation of Polyp Detection Methods in Video Colonoscopy: Results from the MICCAI 2015 Endoscopic Vision Challenge", IEEE Transactions on Medical Imaging - submitted -, 2016

[20] "Precision and recall- Wikipedia, the free encyclopedia." [Online]. Available: [https://en.wikipedia.org/wiki/Precision_and_recall]

7 ANNEX



Annex 1. Project planning