ETIS LAB – CELL TEAM

Development of Computer-Assisted Lesion Detection Algorithms For The Analysis of Videoendoscopy Images and Videos of the Intestinal Tract

Third year Internship Report

Author: Juana Sologuren

July 2023, Cergy-Pontoise

DESCRIPTION

Company name: ETIS Laboratory

Address: ENSEA University, 95000, Cergy-Pontoise, France

Contact: 01 30 73 66 66

Area of expertise: Biomedical Engineering – Computer Science

Tutor: HISTACE, Aymeric

Student: SOLOGUREN, Juana Agustina

Project: Development of a Computer-Assisted Lesion Detection Algorithms For The Analysis of Videocendoscopy Images and Videos of the Intestinal Tract.

Supervision period: 30th of January 2023 – 31st of July 2023

CONTENT TABLE

ABSTRACT

The following report is meant to present an overview of the internship carried out among six months at ETIS laboratory, in specific in collaboration with the CELL team. The main purpose of the project was to apply what was learnt during the previous semester to a practical challenge as the research project on the area of videocolonoscopy image processing. I was conducted by incorporating the use of machine learning methods to achieve the proposed objectives. The report provides a description of the tasks performed among the project, the challenges faced, the decisions and criteria used through it and the conclusions and final accomplishments at the end of it, comparing the first settled objectives for it and the obtained results at the end.

INTRODUCTION

Videocolonoscopy, and others procedure studies of the gastrointestinal (GI) system, are aimed to help professionals in medical procedures to diagnose colorectal diseases. Colon polyps are described by the American Cancer Society as "pre-cancers", as they are most of the time benign but they are a risk for the patient in the future. Therefore, it is extremely important to get an accurate diagnosis while supervision of this kind of lesions in the tract.

There are various kinds of classifications regarding GI polyps, taking into account different characteristics on them. Specifically, in video colonoscopies the first diagnoses that can be concluded is only based on visual characteristics, which entails the possibility of mistaken diagnosis due to the subjectivity of the doctor. Thanks to this, it is necessary to include as much help as possible for these analyses.

In the last few years deep learning techniques had grown their way into the medical world approaches. It is already installed as a useful tool in medical imaging, as is being used to analyze CT-scans, X-rays, MRIs and other types of images. [15]

Although there are already promising methods using ML for detection of GI lesions such as polyps, classification tasks are still not with stablished results. The main goal of this report is to approach a system which can detect and classify polyps by videocolonoscopic pictures using deep learning based methods.

1 LABORATORY PRESENTATION

The project was developed at the *ETIS* laboratories *(Equipe Traitement de l'Information et Systèmes),* working precisely in charge of the *CELL* group (*Research team for Smart Embedded Systems*). The lab is composed by a group of different researchers in various fields of engineering from different academic institutions in Cergy-Pontoise: *Cergy-Paris University CYU*, the postgraduated school of engineering *ENSEA (École Nationale Supérieure de l'Électronique et de ses Application)* and *CNRS/INS2I (Centre National de la Recherche Scientifique)*.

The CELL team is composed by 36 people, among them PhD students and post doctorates. It has achieved to publish more than 150 publications and count with wide national positioning not only among France, but as well has notable international collaboration with other academic institutions among Europe, the US and others. The CELL team is a multidisciplinary team of researchers in various fields as electronics, signal processing and image processing. The principal goals of the CELL group are to tackle the following challenges from an engineering perspective:

- Encourage sustainable telecommunication networks and minimizing their ecological impact.
- Generating intelligent embedded systems, both in hardware and in information processing for different fields of application: advancements in IoT, telecommunications, health, robotics, autonomous cars, and others.
- Create computational embedded systems with heterogeneous embedded computational systems.

Therefore the main goal is modeling and design reconfigurable embedded systems that are reliable and intelligent focusing on the subject of using machine learning for the analysis and processing of data.

1.1 TUTOR

For my participation in the team I was in charge of Mr. Histace, who is a professor for EVE specialization at ENSEA, and also Deputy Director of the school and takes part in the CELL team as the head of it. He does research in Computer Vision, Signal and image processing with main interest in applications among natural science, engineering, medicine and information science. In the *CELL* team his activities are dedicated to smart, reliable and reconfigurable embedded systems for various applications.

Mr Histace has developed research in the matter of data processing and analysis for videocolonoscopy images. He participated with a Spanish colleague Mr. Jorge Bernal. They both participated in the GIANA challenge, a challenge to test different methods for gastrointestinal polyp detections and classification systems, as evaluators and had made various publishes on the topic. Furthermore, they have published a book by their editing named "Computer-Aided Analysis of Gastrointestinal Videos" on their work and on the results obtained on the challenge. The goal of indicating the clinical interests on Gastrointenstinal image analysis systems, then presenting the most remarkable methodologies proposed on the challenge, and also presenting the development and the validation framework used for further future research.

2 THEORETICAL FRAMEWORK

2.1 POLYPS AND COLORECTAL CANCER

Polyps are an abnormal growth of cells in the colorectal cavity, but although these are most of the time harmless, with time can finally evolve into cancer. Colorectal cancer is a extremely usual disease in adults, only in the last year 2023 the American Cancer Society had estimated a number of 106.970 new cases with colon cancer and 46.050 of new cases of rectal cancer only this year. Furthermore, it is sentenced to be the third leading cause of cancer-related deaths for women and men and it is expected to cause around 52.550 deaths this year. [16][17]

As mentioned before, polyps are usually not malignant, but they can be the first sign to future cancer. That is why it is very important for the clinical world to know how to detect potential cancerous-polyps before they evolve and to hystopathologically classify them correctly. Polyps can be in several locations typically like bumps on the gastrointestinal tract and can vary in size. By this and more other characteristics is how they can be classified and detected.

Currently the most popular study for this matter is the colonoscopy, which is a procedure in which the specialist inserts a flexible tube into the rectum and shows the inside by looking at the picture generated by the camera in the tip of the device. This method is very useful for the doctor as it is a good insight to the tissue of the patient. Nevertheless, it is a challenge for colonoscopists to do a correct critic on classifying polyps just by watching their aspect, as the perfect way of getting a good judge is by doing a study on the tissue itself. For example, the differentiation between sessile serrated polyps and hyperplastic polyps is a challenging task for pathologists but anyways important as sessile serrated polyps can potentially develop into colorectal cancer more easily.

2.2 POLYP CLASSIFICATION

There is a wide amount of ways to characterize polyps according different features in them depending on the method used for it, they can be categorized by: their shape, size, how flat they are, if they are sessile or pedunculated, their tissue, how their surface is, between other characteristics. The most common classifications is about the tissue itself and can divide polyps as Hyperplasic polyps, adenomatous polyps and serrated polyps.

Adenomatous polyps have usually a regular appearance, similar to the tissue in the rest of the colon, but with several differences in the growth patterns on them when looking under a microscope. These can be tubular, villous or a mix of the two. The larger in size the polyps are, the more possibilities for them to develop into cancer. Actually this type of polyp is the most cancerous put of all, therefore when this are detected they are highly recommended to be extracted.

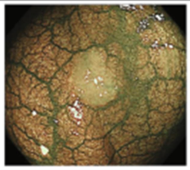
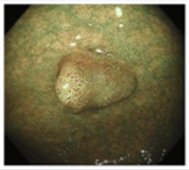
On another hand, Hyperplasic polyps are normally not easily distinguishable from the adenomatous. The biggest difference between them is they are smaller in size and have less potential to become malignant.

Serrated polyps are defined by their saw-toothed appearance and tend to be bigger than the hyperplasic polyps. They are the most hard to detect by endoscopic examination and if they are cancerous they progress quite quickly.

2.3 NICE CLASSIFICATION

Between the classifications the most used while analyzing pictures while endoscopic procedures are the methodologies which are fully about general features in the polyps appearance. The *NICE* classification has its name after the NBI international colorectal endoscopic classification. NBI stands after “narrow-band” imaging, which is nowadays the most frequently used method nowadays in endoscopy. Thanks to this method is able for specialists to classify polyps based on vessel morphology, difference in color among them and the background and surface patterns. By this, doctors can differentiate polyps in 3 different classes:

type 1 Hyperplasic and sessile serrated polyps, type 2 adenoma polyps and type 3 deep submucosal invasive cancer.

	Type 1	Type 2	Type 3
Color	Same or lighter than background	Browner relative to background (verify color arises from vessels)	Brown to dark brown relative to background; sometimes patchy whiter areas
Vessels	None, or isolated lacy vessels may be present coursing across the lesion	Brown vessels surrounding white structures**	Has area(s) of disrupted or missing vessels
Surface pattern	Dark or white spots of uniform size, or homogeneous absence of pattern	Oval, tubular or branched white structures** surrounded by brown vessels	Amorphous or absent surface pattern
Most likely pathology	**Hyperplastic & sessile serrated polyp (SSP) *** **	**Adenoma**** **	**Deep submucosal invasive cancer**
Endoscopic image			

* Can be applied using colonoscopes with/ without optical (zoom) magnification

** These structures (regular or irregular) may represent the pits and the epithelium of the crypt opening.

*** In the WHO classification, sessile serrated polyp and sessile serrated adenoma are synonymous.

**** Type 2 consists of Vienna classification types 3, 4 and superficial 5 (all adenomas with either low or high grade dysplasia, or with superficial submucosal carcinoma). The presence of high grade dysplasia or superficial submucosal carcinoma may be suggested by an irregular vessel or surface pattern, and is often associated with atypical morphology (e.g., depressed area).

Image 1. NICE classification board [11]

2.4 PARIS CLASSIFICATION

The Paris classification is also used in endoscopic treatment for early carcinoma on the GI tract. It was based in the earlier Japanese classifications of superficial lesions on the GI system, and it is done in two stages. Primarily it assesses the endoscopic appearance, defined as type 0-I as “Polypoid” and as 0-II or 0-III “Non-Polypoid”.

Type 0-I is referred to polyps that are protruded in some way, 0-Ip for protruded, pedunculated as the 0-Is protruded, sessile. The type 0-II is referred to flatter polyps, 0-IIa is for slightly elevated polyps, 0-IIb is for flat polyps and the 0-IIc is for slightly depressed polyps. The type 0-III is for excavated polyps, more depressed shape polyps.

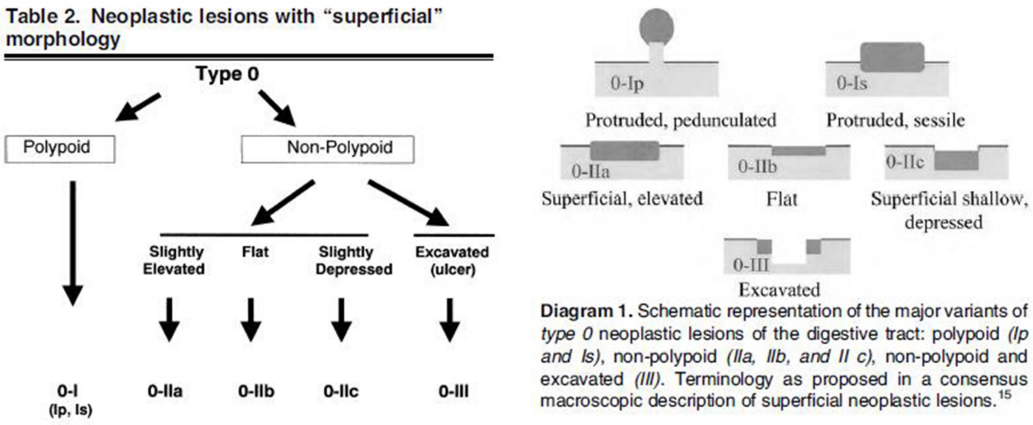


Image 2. Paris classification for polyps [19]

2.5 IMAGE PROCESSING: VIDEOCOLONOSCOPY AND MACHINE LEARNING

In the last few years the use of machine learning applied to various fields has been growing among the scientific community. Thanks to its performance and results, it has been very helpful for optimizing time and processing to do several activities regarding image processing. It was not an exception with medical images, which they are a complex and still important to be analyzed correctly, as most of the time the conclusion made from a patient image is elemental for their treatment and health. Evermore, it can define the destiny of their lives.

In the case of GI lesions, more specifically for polyps, the research on detection stated around the 2003 with the first methods of polyp classification using handcrafted methods. These utilize local features of the images, such as histograms, pattern descriptors and others, to localize polyps in the images. These kept growing and during the years, but there was a downpoint to the methodology: this were good at solving an specific problem but not in a general term, as they usually do not generalize well across multiple datasets. Around 2016 methods involving machine learning started to emerge and were showed to have better results.

Nowadays machine learning is already being used in medical images for research and little by little it started to be introduced in the use of professionals. As I

mentioned in the previous chapter, ETIS professor Mr. Histace has participated as a jury in the GIANA challenge during the 2017 and 2018. During these it was proposed to participants to present methodologies for detecting and classifying polyps. They had been successful in obtaining good results in classification matters using certain segmentation approaches. They also proposed a challenge on classification matters for polyps in which there had been several attemps with good results, but even today it is still growing in its discovery. Then the clear objective of this project was to do research in the relevant challenge about polyps: how can machine learning tools can be used to classify medical images taken during colonoscopy procedures.

2.6 SOFTWARE

For the development of the project I used various software tools that made easier the process of programming. Every code was made in python 3.19.13, and most of it were written in Jupyter Notebooks and run on Visual Studio Code 1.79.2. Jupyter notebooks are useful as they are an open-source project that lets the user easily combine markdowns and executable python code in one notebook. It is friendly for the user, a clean way to code helping very easily to catch problems in the coding. As well I used miniconda version 23.3.1. The miniconda is a software distributor used in data science to simplify package management and deployment. This was used for creating safe environments to run most of the codes on them, and also used its terminal to run some code programmed.

The GPU used was NVIDIA GeForce GTX 1660 Ti, CPU was Intel Core i7-9750H, with an available RAM of 8GB an the operating system was Windows 10 64-bits.

The specific libraries used in every specific staged are detailed in the following chapters.

3 ACTIVITIES PERFORMED

From the first point of the working period the main objective of the internship was settled: generate a system that could detect and classify polyps. Basing myself on some of the bibliography mentioned in the pertinent articles about the challenge I started to get myself into the classifying used methods and the challenges in the future regarding them. The workflow was then proposed.

3.1 SEARCHING FOR DATABASES

The starting point for coding and testing different algorithms was to get databases. I began by looking at the databases mentioned and used during the GIANA challenge, which were reliable as they were already used for deep learning method purposes. The found databases were:

Database name	Number of pictures	Mask	Classification
CVC-ClinicDB [20]	612	612	N/A
Kvasir-SEG [21]	1000	1000	Sessile Serrated*
Sessile-Kvasir-SEG	196	196	Sessile polyps
Dataset by Mesejo-Pizarro [22]	76	N/A	Adenoma, Hyperplasic and Sessile Serrated
CVC-ColonDB	379	379	N/A
ETIS-LaribPolypDB [23]	196	196	N/A

Table 1. Databases used during the project

3.2 DETECTION

3.2.1 U-Net APPROACH

The following stage was to focus on finding a detection system for the polyps. Through deep investigation I discovered that the most used techniques in detection in the field used image segmentation algorithms. Image segmentation is basically a method based on dividing an image into meaningful and distinct regions and analyzes them separately. This is helpful to border specific objects in images, in this case distinct polyps from the background, which would make it simpler to extract specific features and characteristics.

Deep learning models have shown excellent results for image segmentation in this field of study. As most of the data bases obtained have their ground truth (mask pictures with outlined shape of each polyp), I would be capable of training and validating models proposed.

To implement the polyps segmentation I chose to use a U-Net architecture. The U-Net is a fully convolutional neural network that was specially developed for biomedical image segmentation. It is called after the letter U, as it follows a symmetric architecture shaped liked it: It starts with pictures of 256x256 pixels going through convolutional layers followed by a maxpooling layer, repeat the same layering until it downsizes to 16x16pix. Then it starts upsizing the pictures by getting this result trough several layering made by convolutional layers again but followed up by up convolutional layer each time until the picture is finally shaped 256x256.

For the implementation of this approach among the most significant libraries used for deep learning functionalities were the *keras V.2.11.0*, several callbacks are used EarlyStopping, ModelCheckpoint, ReduceLROnPlateau, CSVLogger, TensorBoard, Sequence. The train_test_split function on the *sklearn.model_selection* library was used for dividing the dataset into the training and the testing set for the CNN.

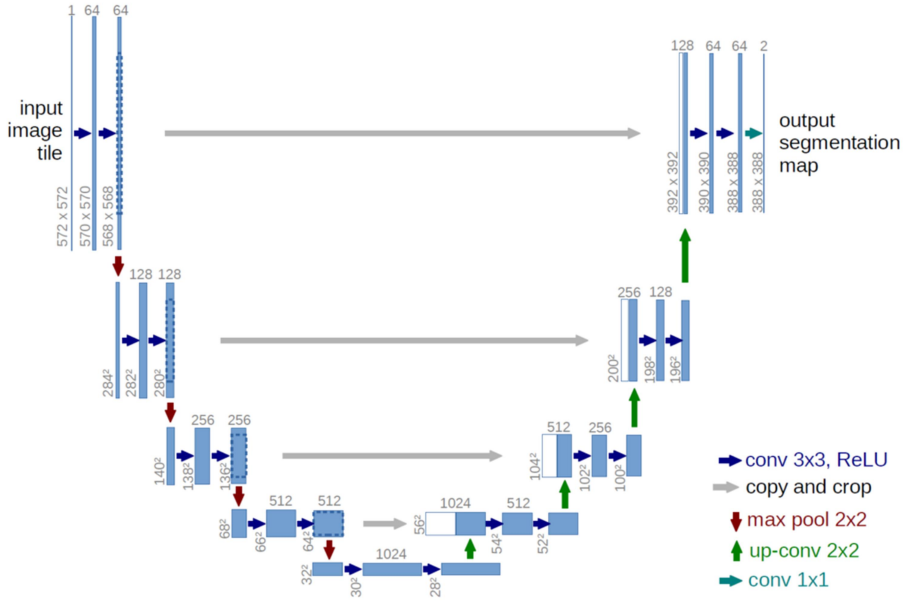


Image 3. U-Net architecture scheme.

This architecture was tested using all of found databases except for the dataset without the ground truth masks. The results for training and testing on the same datasets were at first promising, but at the time to use as training certain DB and changing to test with other one result were not important. Best outcomes were taken by training with a mix dataset made out of Kvasir and CVC-Clinic datasets. However, after various mix and matches of databases for learning and testing, I could not generate enough good masks to analyze the pictures. The segmented area classified as 'polyp' was not capturing the entire lesion, so as a consequence some information needed would be lost. Finally while it showed potential, the limitations led me to explore other strategies in to the detection methods ensure the polyp is wholly detected correctly, loosing details about shape and outlined of the polyp.

3.2.2 YOLOV7 APPROACH

Other popular approaches using CNNs used in the medical field for image processing is the yolo architecture. *YOLO* stands up for You Only Look Once, architecture that deals with segmentation as an object detection problem. This program divides the input image into a grid and, by an already trained system with labeled images of the sort, assigns bounding boxes to objects identified in the images. This architecture efficiently detects and labels the result pictures in them as its prediction.

Since YOLOv3 and YOLOv5, were previously algorithms used for detection I opted for a more recent version using yolov7, as to implement a similar approach and obtain a faster performance. To implement this model I started by selecting pictures which were best to use in the training and of the CNN manually. To achieve a more integral and robust system I selected pictures from all the disponible datasets: in total there were 415 pictures to train the model; 76 images from the classified database; 89 images from ETISLarib dataset; 99 images from the CVC-ClinicDB; 151 images from the Kvasir-SEG database, and there were 50 pictures used for the validating, all of them from the Kvasir. Then, I labeled each picture using the labelImg Pyton tool. By this I could annotate the polyps for each training picture and mark its boundaries with a tag. The label is saved in different .txt archives for every picture in another folder chosen by the user.

Afterwards, I trained the system using these pictures and labels. A custom detector for the specific task was created and was then used to test the system. To evaluate the effectiveness of it I tried the system to see the results with several mixed groups of pictures. It was observed that the results were poor, as very little amount of polyps were detected. To solve this problem I changed the original threathold 0,5 in order to achieve which one suits the best for the system: detecting enough polyps without making too much mistakes.

After achieving a sufficient outcome in the detection, I could generate new pictures from the segmented polyps detected and save them for their use in the classification. These are useful not only for simplify the accurate classification task

in the next step of the project, but also as an effective first filter in the processing of images.

For this approach the remarkable libraries used were matplotlib V. 3.2.2 , requests V.2.23.0, spicy V.1.4.1 for specific functionalities. For logging functionalities I used the tensorboard library V.4.41.0, for plotting and data visualization the pandas V.1.1.4 and the seaborn V.0.11.0 libraries. Also I utilized PyTorch version 1.12.1+cu113 as a deep learning framework, installed with CUDA support. Including some PyTorch libraries utilized for the implementation were the torch.nn to define the CNN architectures, the torch.utils.data to manipulate the data, between others. Besides, for model loading and creation the models.experimental and models.yolo were used. for data handling utils.datasets, computation of loss and plotting utils.loss and utils.plots.

3.3 CLASSIFICATION

One of the major challenges of the project was encountered at the time of thinking what strategy could be used for the classification. The limited availability of databases with polyps classification was a significant obstacle as it was a limitation at the time of choosing with which medical classification I could work for my system.

After some research on different classifications I decided to base the classification method on the NICE classification (mentioned in the previous chapter). Due to the fact that that was the only database that allied the pictures with the classification. Although the dataset, has some disadvantages as the few quantity of pictures it has, and the low quality of them, it emerged as the only viable option on the resources.

The NICE classification is mainly based on easy-to-sight features, so it was a great approach as for me to understand and create a program which could evaluate only by simple images of the lesion. While it may not be of a high complexity, it provided a practical workaround given the constraints imposed by the lack of suitable databases with comprehensive classifications.

3.3.1 GLCM TEXTURE ANALYSIS

Using the NICE classification the key features to consider during the analysis were the color of the polyp, the distinctness and homogeneity of the lesion in comparison with surrounding GI tract, the presence and amount of vessels surrounding the polyp, and the texture in the polyps surface. All of these are easy features to detect by an image of the GI lesion.

The first approach was focused on making a system which could classify the polyps according the texture on the surface of the polyp, as in the images already classified this was the most distinct characteristic between them all.

After several strategies considered the tutor of the project suggested using a co-occurance matrix for the analysis of the pictures. GLCM stands for Gray Level Co-occurance Matrix and it's a widely used technique for texture analysis applications. Its concept was firstly introduced in 1973 by Haralick, who defined the glcm as a square matrix G of order N where the (i, j)th entry of G represents the number of occasions a pixel with intensity I is adjacent to a pixels with intensity j. This adjacency can be defined to take place in each of the four directions from the pixel (in the horizontal, vertical and diagonal directions). The texture features could be calculated for every pixel by averaging the four directional co-occurance matrices.

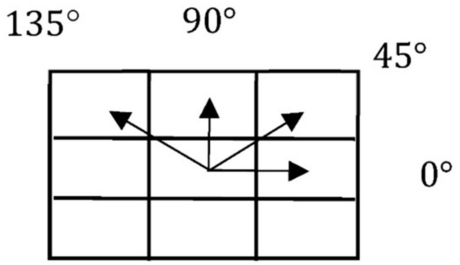


Image 5. Co-occurance matrix explanation Haralick [24]

Considering an image in gray scale level, each pixel has a value from 0 to 256 describing the brightness of the pixel, and the difference between the color of the pixel and its neighbors can then define and describe texture characteristics. As more different are the neighbor pixels from each defined pixel, the most texture we can find through the polyp.

The trace of the normalized co-occurance matrix is proposed as a new feature to identify constant regions in an image. The matrix is made by summing the values along the main diagonal of the matrix. As the GLCM is symmetric the diagonal is going yo represent the pairs of pixels with the same intensity value next to each other, the higher the value the most homogeneous are the pixels.

3.3.2 FIRST APPROACH

The first approach using the GLCM consisted of several steps. I took the pictures from the training set and convert them to grayscale. Then each picture was subjected to a function in which the GLCM is calculated using the graycomatrix function from the scikit-image library every time switching the distances and angles to obtained the most information possible. When calculating the matrix for different values of distance (distance between the pairs of pixels considered in the GLCM calculation, the distance from the pixel to the neighbor pixels taken into account) and angles (the direction in which said pixels are considered) it is possible to analyze different texture aspects, it provides an insight to spatial relationships and patterns of the image.

Various features can obtained by the graycomatrix function, in this case the analyzed ones were, energy (which quantifies the homogeneity of the texture), correlation (which describes at which extend a pattern is defined), dissimilarity (the difference in intensity between pixels), homogeneity (measures the homogeneity in intensity of the values per regions) and contrast (measures the variations in intensity in neighbor pixels) (Annex 1). The features were calculated for the following distance-angle values: for angle 0 distance 1, 2, 3, 4, 5 and for angle 0, $\pi/4$, $\pi/2$.

By the LightGBM Python library I implemented the Light Gradient Boosting Machine algorithm to process the information taken by the previously explains function to extract features. This is a type of gradient boosting decision tree based algorithm and is usually used for classification, as it provides accurate and remarkably fast results for machine learning tasks. This is another reason which is

a good choice to use it, as it is the most convenient choice for quick processing of large datasets.

The system was trained and tested using the previously cropped images of the Mesejo-Pizarro database pictures with the polyps segmented from 2 classified polyps: 66 images for adenoma and 34 images for Hyperplasic. To achieve a better and robust classification other images were added. From the endoscopy campus article we extracted 9 images for hyperplasic polyps, from which by the yolov7 system were filtered to 6, and 11 images for adenomatous polyps, from which by the yolov7 system were filtered to 9. Summing up, the total of used images in the system was 115: adenoma pictures for training used were 61 and 14 for testing, and the hyperplasic pictures were 32 for training and 8 for testing.

For this approach the utilized different libraries, for machine learning tasks I made use of the lightgbm V.3.3.5 library for the gradient boosting model. The parameters used for the model were: learning rate=0.05, number of leaves 1000 and maximum depth=100. For data analysis I used various libraries, the most important being the pandas V.1.5.1 and the sklearn.metrics, used for calculating and graphic *precision_score, recall_score, f1_score, confusion_matrix and precision_recall_curve*. From the scikit-image V.0.20.0 library I used the models sobel from the skimage.filters for edge detection and the graycomatrix and greycoprops from the skimage.features library for calculating the glcm features.

3.3.3 SECOND APPROACH

With the intention of leveling up the system made based on GLCM I tried to find another approach of the previously made program. The system was based on a previously created algorithm implemented in the research for another system made for breast cancer classification of images [26]. In the quoted paper the premise was to work with the GLCMs taking a pretrained model (in their case DenseNet201) and extracting deep semantic features of the last dense block from part of the convolutional layer. Then a SVM system support vector machine is used for classification. Also they tried with more classification models and compared them.

As the features aimed to classify in this project are made on basic to sight characteristics based on texture, color and edges, the used method is by extracting deep features from CNNs. A deep neural network is a deep learning technique that consists of a feeforward network made from several hidden layers. By using pre-processed techniques to extract the features from the images and fee those features as imput of a classifier to evaluate and do a final prediction we are processing features like texture.

For the classification I used the SVM as they implement in the breast cancer system. Furthermore, in order to make a wider analysis and to test if better results could be achieved, I researched more about what other classifiers could be built for the polyp classification. The other statistical methods for classifying were logistic regression (statistic method based on predicting the probability of an event occurring such as voted or not voted based on given dataset of independent variables), Random forest (system based combining the output of multiple decision trees), Gradient Boosting (method based on transforming weak learners into stronger ones taken from decision tree models). The classifier models were implemented using the scikit-learn library V.0.21.0. and the classes used were: *LogisticRegression, RandomForestClassifier, GradientBoostingClassifier, svm.*

To implement the main idea I analyzed which different CNNs I could used for the system. As they utilized in the breast cancer project I used DenseNet201, a densly connected-convolutional network 201 layer deep based on taking previous outputs from the model as inputs for future layers, the ResNet50 and ResNet152V2, which are residual networks made from residual units or blocks which skip certain connections one with 50 layers and the other one with 152 layers, the inceptionv3 which is a method with multiple parallel convolutional layers of different sizes, the MobileNetV2 which uses a CNN of 53-layer-depth known as depthwise separable convolution, VGG16 16-layer-depth CNN algorithm and Xception a 71-layer-depth CNN. The models were all implemented by importing the pretrained models from the *Tensor Flow Keras* V.11.0 library: *DenseNet201, MobileNetV2, ResNet50, ResNet152V2, VGG16, Xception.*

After long research the implementation of the scheme was executed by creating a code in which as first step the images were divided by three channels according to

each rgb-colorscale, and from each of them extracting the GLCMs for distances values in between [1, 2, 3] and angles [0, π/4, π/2, 3π/4]. For each of these the features calculated were contrast, correlation, dissimilarity, energy, homogeneity and ASM. To obtain these features and the glcms the graycomatrix and graycoprops functions from the keras library were used. Besides, the images go through a CNN, mentioned previously, and certain deep layers from them are extracted. Afterwards, these two are merged and submitted to a classifier method, also already mentioned. By this last step, a prediction of which category the polyp is between adenoma and hyperplasic is can be generated.

Other important used libraries in the implementation, besides the ones mentioned in the previous section for glcms, were the *matplotlib* V.3.7.1, the *sklearn* V.1.2.1 for other machine learning .metrics for calcules and graphics using the functions: *precision_score, recall_score, f1_score*, *roc_curve, auc* for the sklearn other functions as the *train_test_split* function to split a dataset into validation and training, from *sklearn.metrics* the *accuracy_score* function and the confusion_matrix function for computing results, lastly the Label Encoder function from the *sklearn.preprocessing*.

4 RESULTS

In this chapter the results obtained by the utilized systems explained in the previous chapter are presented: for the detection task the YoloV7 results and for the classification task the 2 implementations using GLCMs.

4.1 DETECTION

As mentioned before, the Yolov7 system was trained and evaluated using a custom-made dataset using pictures from all datasets used, as mentioned before. In the testing of the system a confusion matrix was computed, showing the True Positives (TP), in this case referring to polyps correctly found, True negatives (TN), for correctly classified background, False Positives (FP), wrongly classified section as polyps, and False Negatives (FN), wrongly classified section as background. The metrics utilized on the system testing were the Accuracy(A), Precision-Recall Curve(PR), F1-Score (F1), Precision (P) and Recall (R). (Annex 2). The graphics obtained during the training-validation of the system were

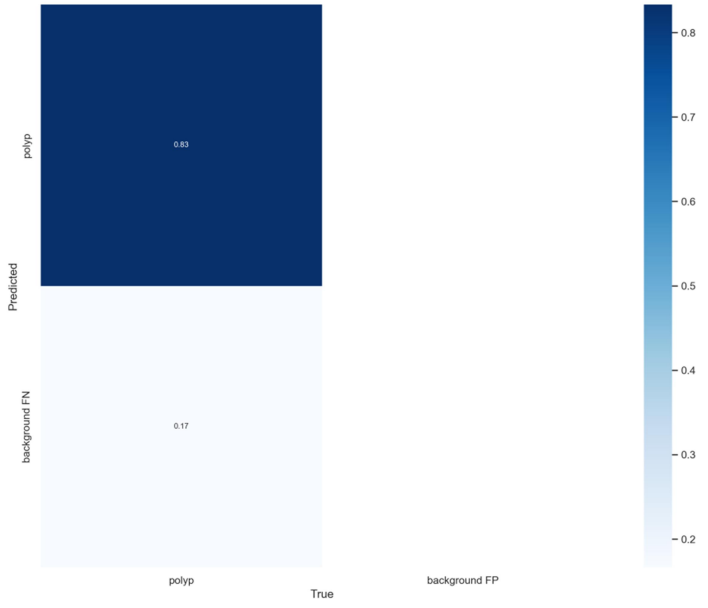


Image 6. Confusion matrix. The calculated TP=0,83, TN=0, FP=0, TN=0 and FN=0,17. The accuracy calculated was 0,83

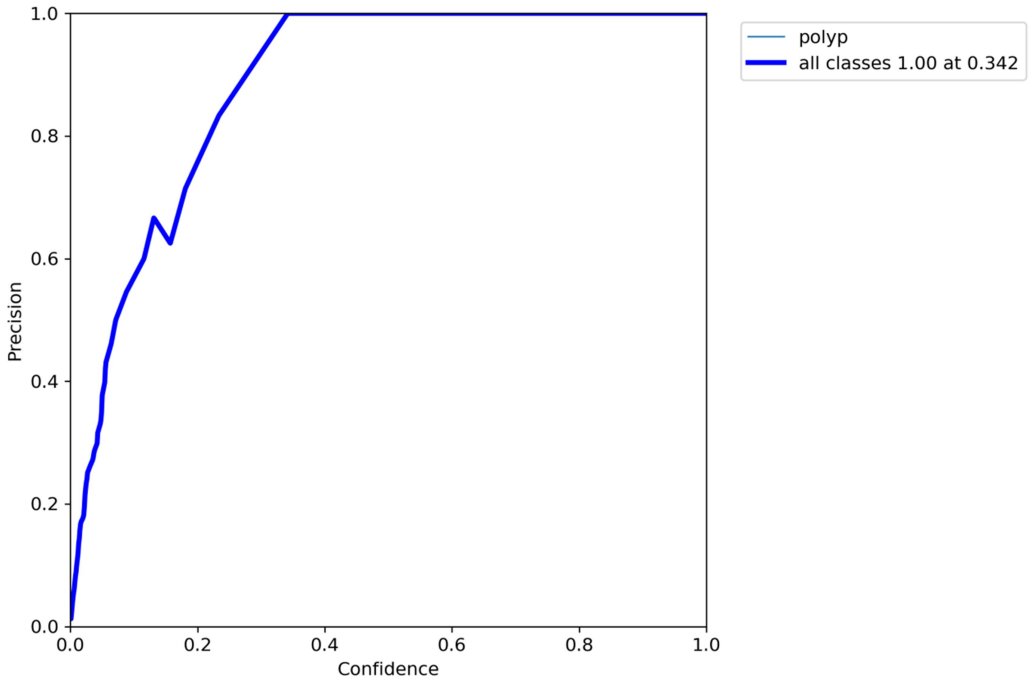


Image 7. Precision curve in reason of confidence

As the confidence level increases, the system is more rigorous to detect positives in the processing of the images, therefore the amount of positives detected gets lower but with more confidence in the prediction, so its more rigorous to make a positive prediction. When the confidence is low we have more possibility for the system to detect positives, wrong or right, between these we have True Positives (TP) and (FP). As precision value gets higher means that the amount of true positives predicted increases. The graphic shows this tendency, as confidence level increase the TP increases until the value of 1, this means that after 0,342 of threshold there is no more FPs detected. Some positives may be lost as confidence increase as well, the FN will increase along with this value.

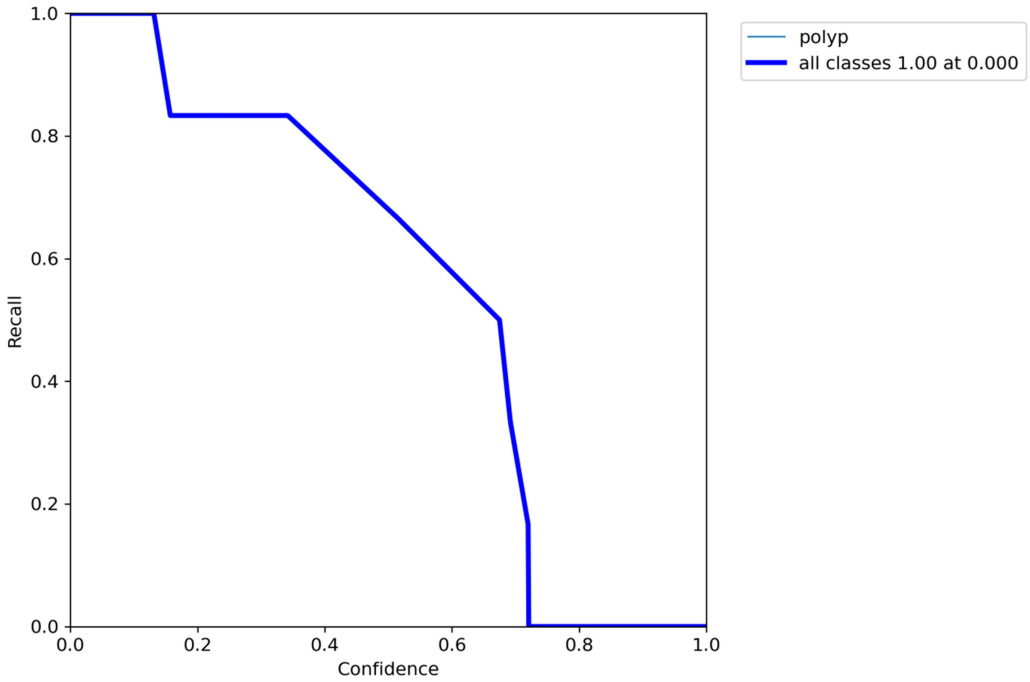


Image 8. Recall curve

As in Annex 2, the recall value represents the predicted positives in reason of the ground truth positives total. At a lower confidence level the system allows itself to do more positive predictions, which concludes in a higher TP rate but also means a higher value of FP. We can see this in the graphic, for low confidence levels recall is at maximum, as it detects positives easily. When the confidence level starts increasing there is less TP values, as less positive predictions are done. By a threshold value of approximately 0.4 we can see that the recall decreases quickly which means that the system is going to detect less and less polyps.

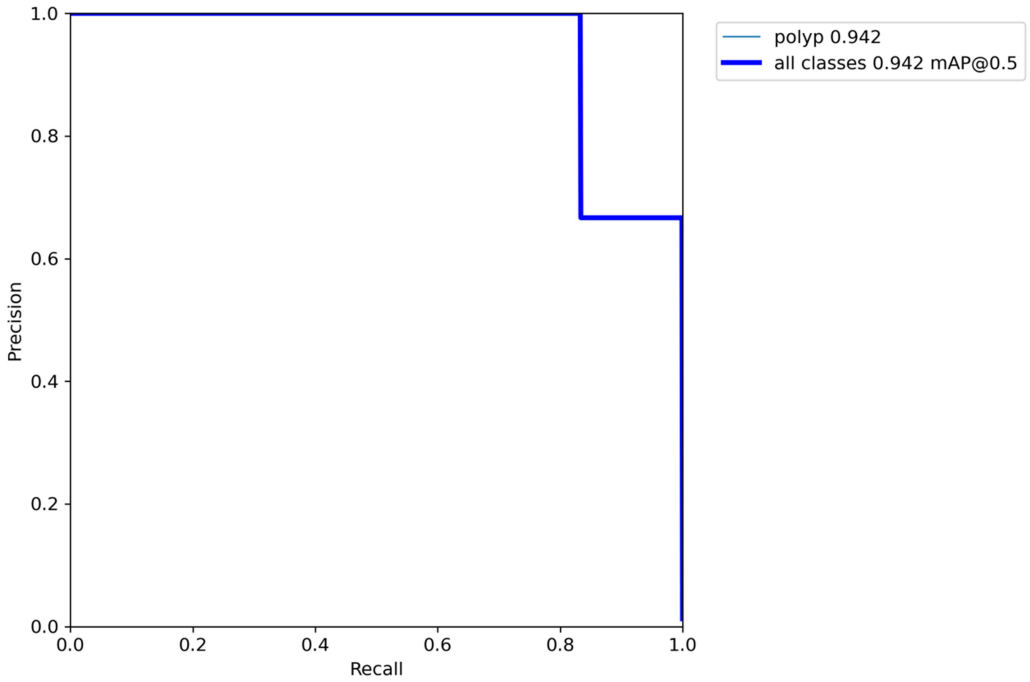


Image 9. Precision-Recall curve

This graphic shows the realtioinship between FP value and FN. At highest recall the predictions to be positive increases, increasing the TPs but also the FP. This lowers the precision value as the predictions to be positive have lower accuracy. For high precision values there are more TP predicted and lower FP, higher FN. This results in a lower recall, as the probability for the system to predict positive decreases.

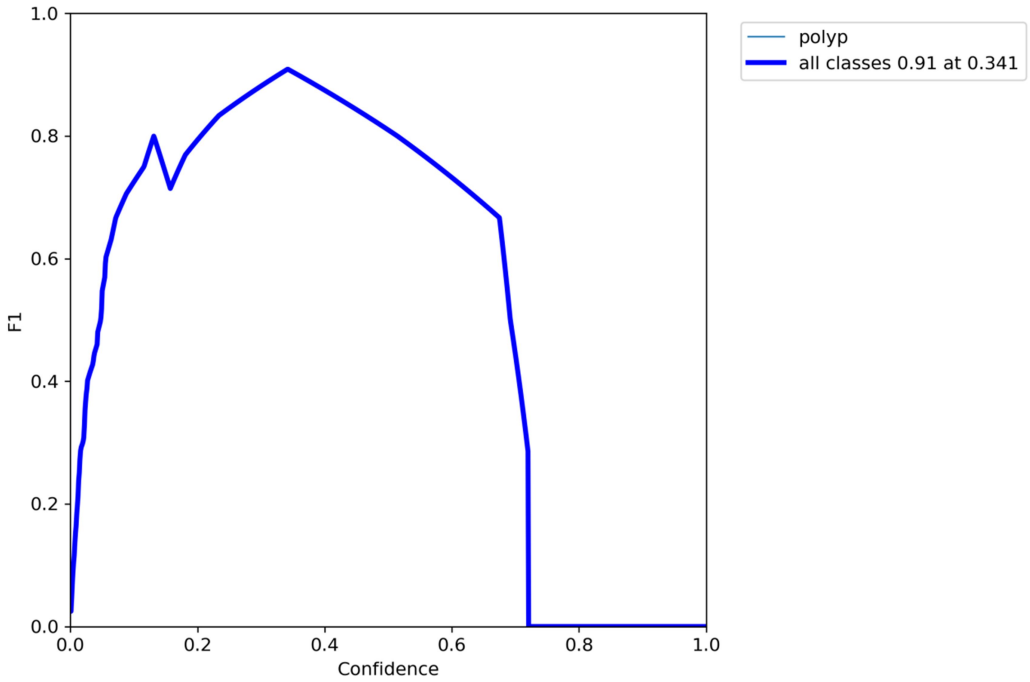


Image 10. F1 Curve

The F1 curve in function of the confidence level shows the relationship between the probability or level of certainty assigned by the model to its predictions and the overall measure of the model performance. We can see here that as values of confidence start to increase the system gains better accuracy in predictions, so the F1 score increases. This occurs as the number of predicted positives decreases, predicting more accurately, the precision increase and the recall decrease. At one point of threshold or confidence the graphic starts to lower its value, getting a peak value as optimal for the system. This happens as the positives predictions decrease and elevating FNs, reducing sensitivity. So we would get the best value for threshold around the F1 peak score, which would be the balance between precision and recall.

The optimal balance value in this system is F1 = 0,9 at 0,341 of confidence threshold. This proved the most efficient value for the system to work. Although this value, I decided to work with a 0,2 threshold to detect polyps, as the dataset classified needed to be segmented for the next stage of the project was very limited I needed the most polyps possible to be detected.

The test batch from the validation set of pictures resulted:

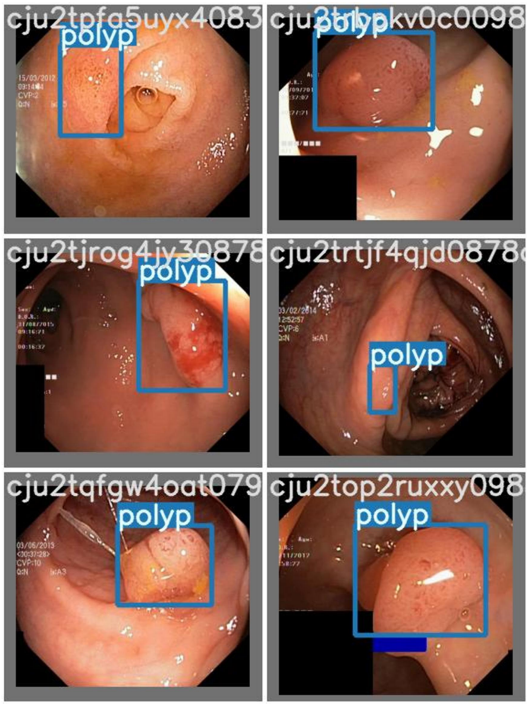


Image 11. Polyp labeled ground truth pictures

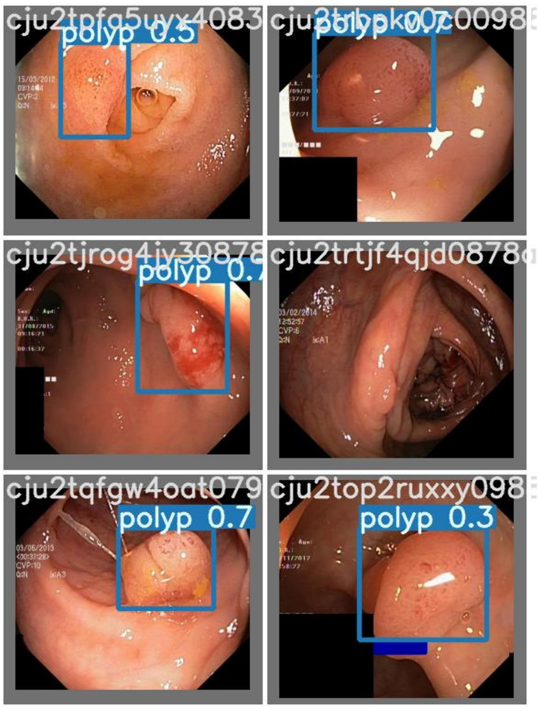


Image 12. Polyp detections predicted by YoloV7

Some examples of the results obtained using this model is presented below, for correctly and wrongly detected polyps.

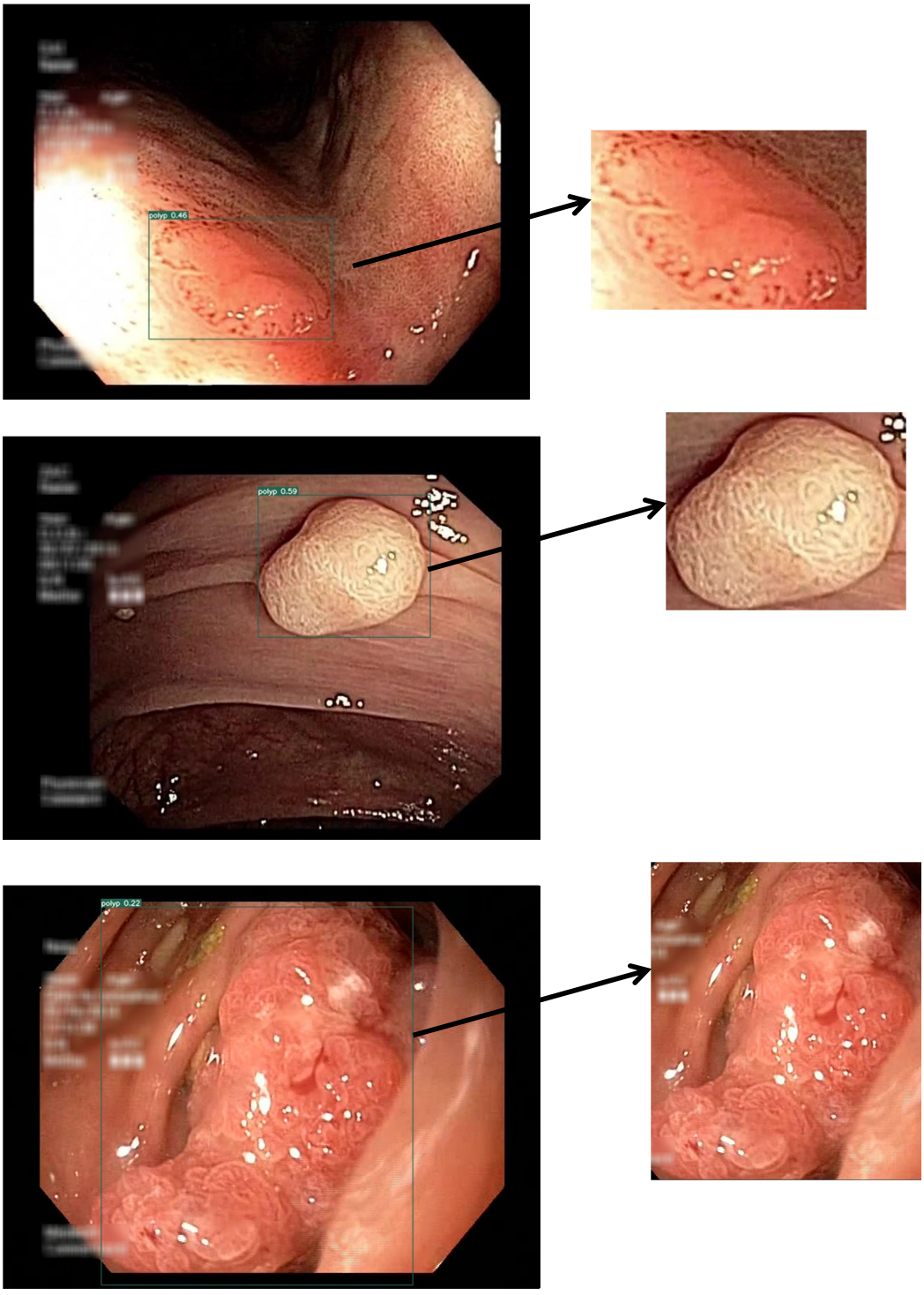


Image 13. Correctly detected adenomatous polyps and the segmented image obtained.

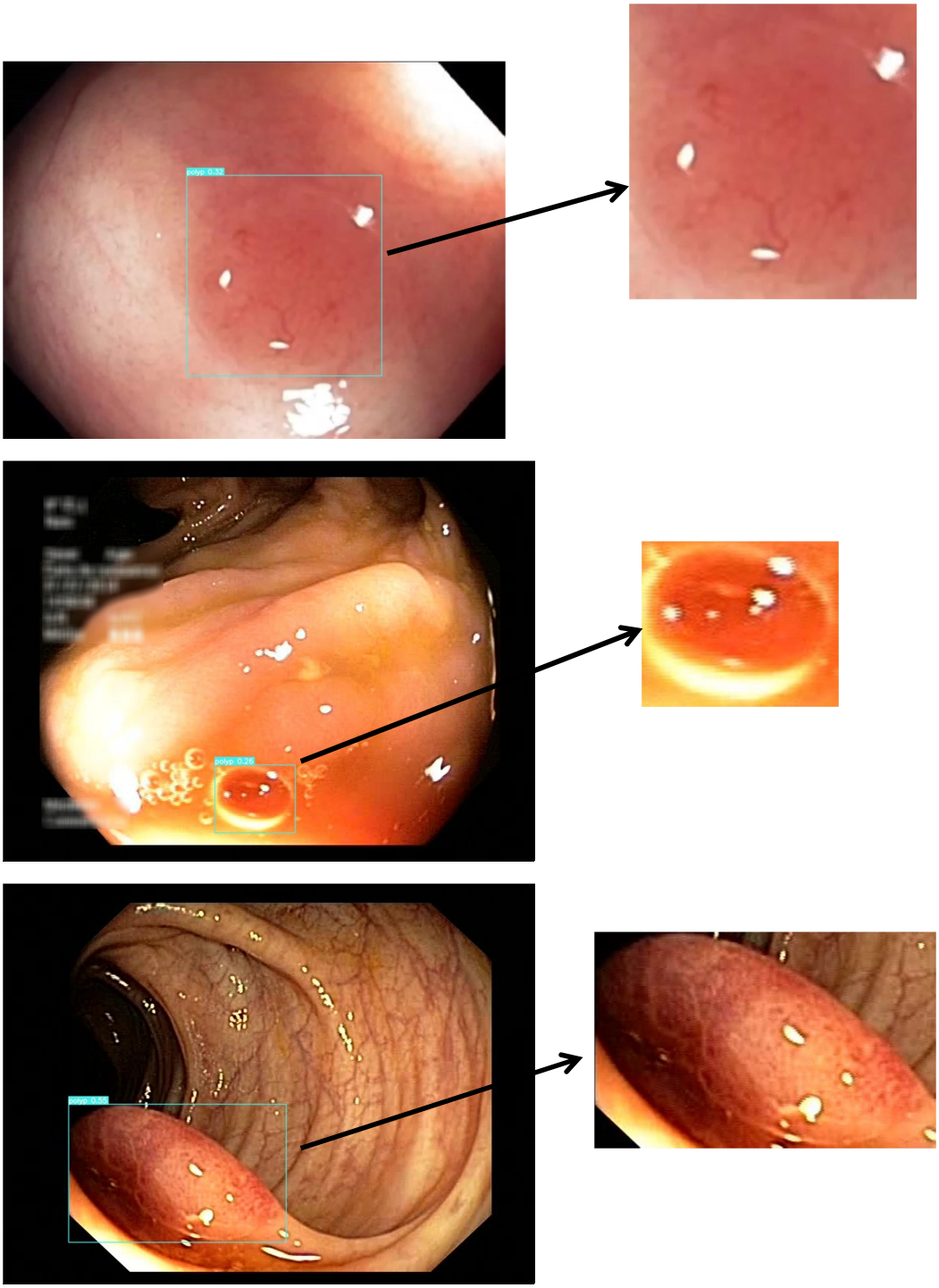


Image 14. Correctly detected Hyperplasic polyps and the segmented image obtained.

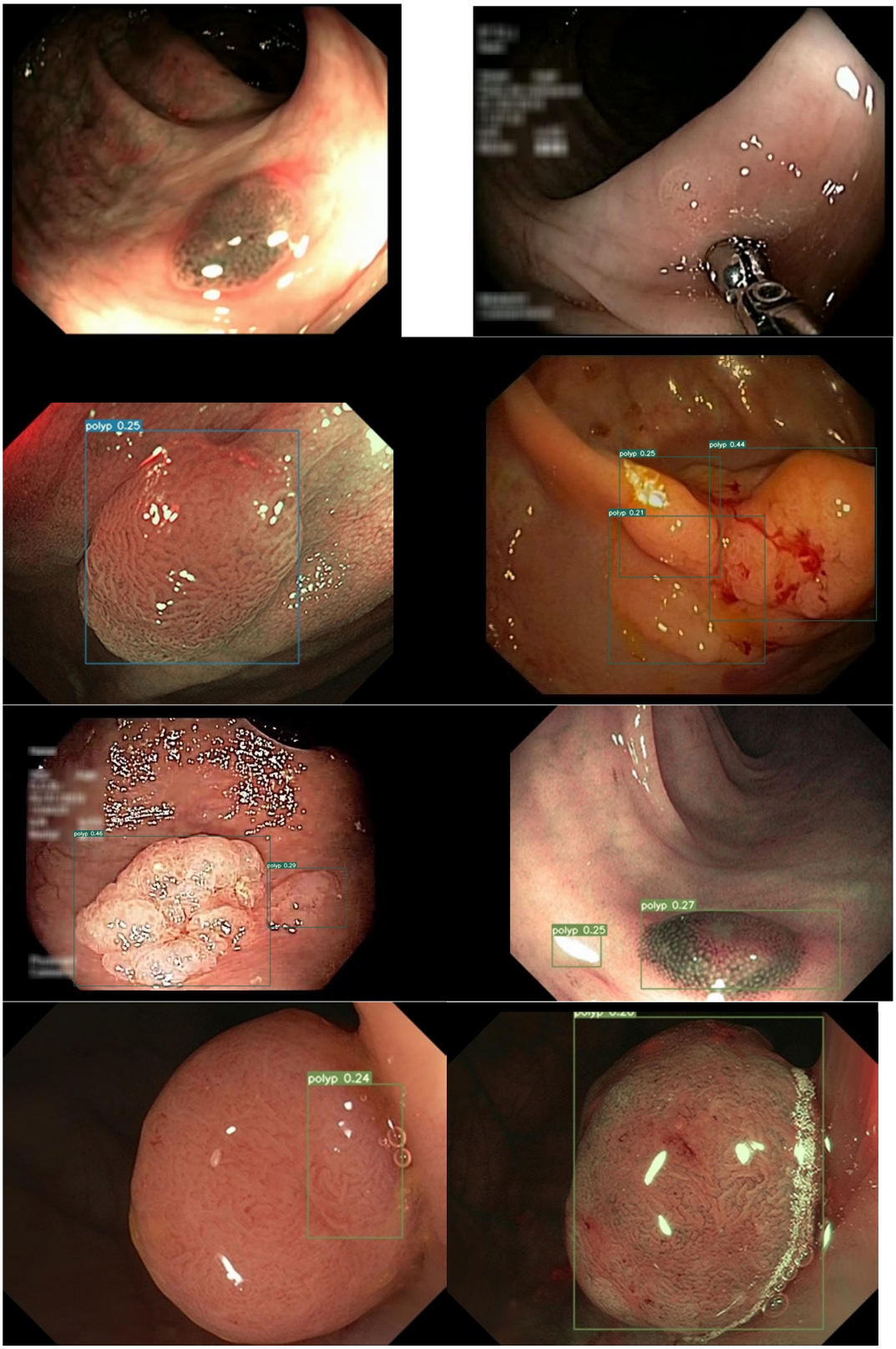


Image 15. Misdetections on polyps examples

4.2 CLASIFICATION

For the classification stage of the research machine learning was applied to texture by extracting GLCM features, using the methodology explained before. The obtained values for the GLCMs extracted and the process with the lightGBM model were the following.

Extracted GLCMs		Accuracy	TP	TN	FP	FN
distance	angle					
1	0 - π/4 - π/2 - 3π/4	0.773	4	13	1	4
1-2-3	0 - π/4 - π/2 - 3π/4	0.773	4	13	1	4
1-2-3-4	0-π/4- π/2	0.682	3	12	2	5
1-2-3-4-5	0	0.727	4	12	2	4
1-2	0	0.727	4	13	1	4
1-2-3-4-5	0 - π/4 - π/2 - 3π/4	0.682	3	12	2	5
1-2	0 - π/4 - π/2 - 3π/4	0.773	4	13	1	4

Table 2. Accuracy and confusion matrix values taken varying the distance and angles.

The best performance was given by the measurements for the distance pixel values from 1-3 and in the angles 0 - π/4 - π/2 - 3π/4. The best performance of the system was obtained using the parameters on the last row of the table, the results for these is:

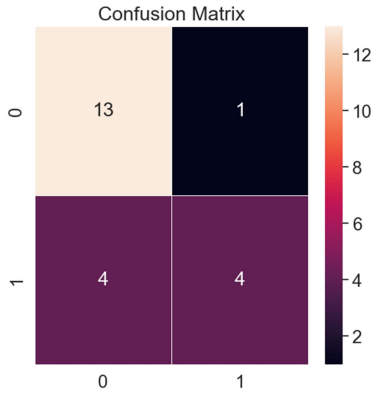


Image 16. Confusion matrix. The calculated TN=13(true adenoma), TP=4 (true hyperplasic), FP=4 and FN=1

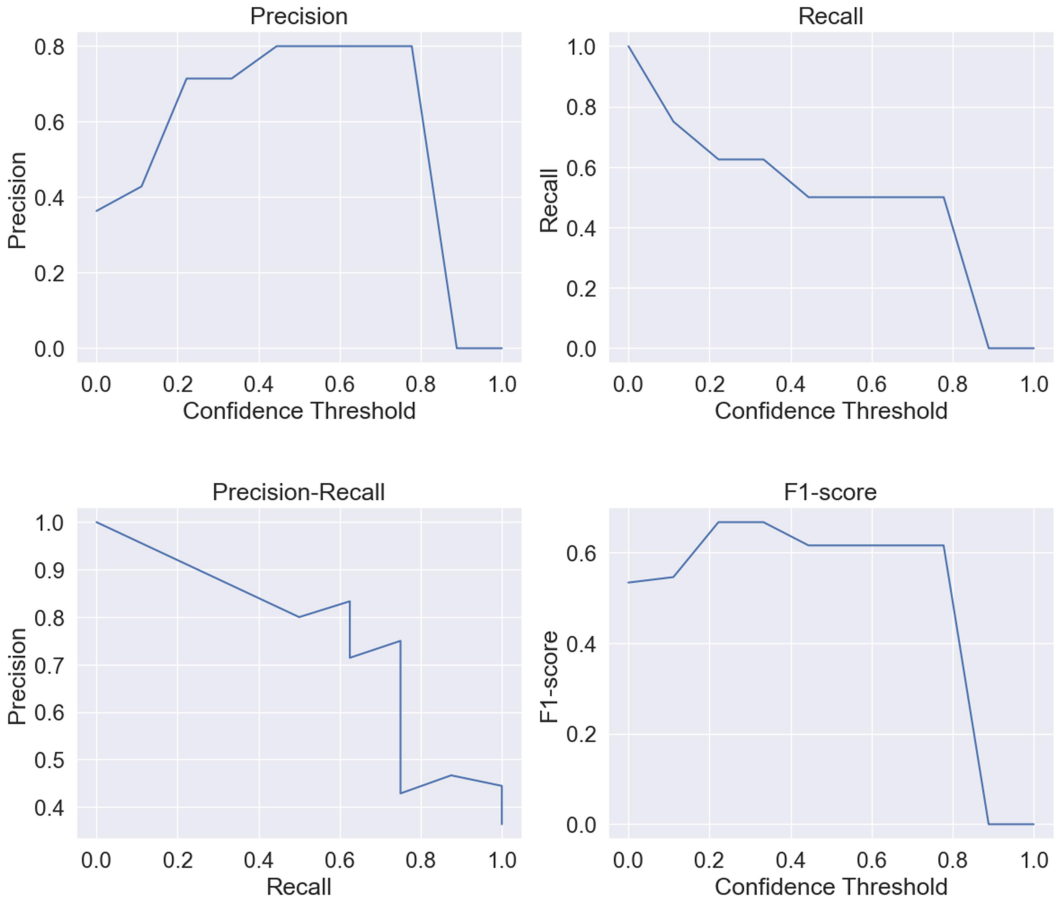


Image 17. Confusion matrix. The calculated TN=13(true adenoma), TP=4 (true hyperplasic), FP=4 and FN=1

Here we can see that the best precision and recall were approximately between the values 0.2 and 0.3 of threshold. We can see that as the confidence increases the precision increases as the accuracy of the prediction is better. For the Recall as the confidence increases it decreases as fewer positives are being detected. In the threshold mentioned range we get the most accurate predictions, so the systems detect a good amount of TP (high) and FN (low). As said before for the threshold values mentioned the system is the most efficient, being the most accurate at predicting accurately positive and negatives. For precision around 0,85 and Recall 0.60 we have the best values.

For the second approach for the classification the results taken from the implementation were:

Models used for extracted deep features	Classifier model	Accuracy	TN	FP	FN	TP	Precision	Recall	F1	Specificity	Sensitivity
DenseNet201	SVM	0.783	13	5	3	2	0.625	0.714	0.667	0.812	0.714
	Gradient Boosting	0.565	11	5	5	2	0.286	0.286	0.286	0.687	0.286
	Logistic Regression	0.696	10	6	1	6	0.500	0.857	0.632	0.625	0.857
	Random Forest	0.826	15	1	3	4	0.800	0.571	0.667	0.937	0.571
InceptionV3	SVM	0.783	13	3	2	5	0.625	0.714	0.667	0.812	0.714
	Gradient Boosting	0.435	7	9	4	3	0.250	0.429	0.316	0.437	0.429
	Logistic Regression	0.739	12	4	2	5	0.556	0.714	0.625	0.750	0.714
	Random Forest	0.739	15	1	5	2	0.667	0.286	0.400	0.937	0.286
MobileNetV2	SVM	0.696	12	4	3	4	0.500	0.571	0.533	0.750	0.571
	Gradient Boosting	0.783	13	3	2	5	0.625	0.714	0.667	0.812	0.714
	Logistic Regression	0.65248	12	4	4	3	0.429	0.429	0.429	0.750	0.429
	Random Forest	0.739	12	4	2	5	0.556	0.714	0.625	0.750	0.714
ResNet50	SVM	0.826	14	2	2	5	0.714	0.714	0.714	0.875	0.714
	Gradient Boosting	0.652	11	5	3	4	0.445	0.571	0.500	0.687	0.571

	Logistic Regression	0.826	13	3	1	6	0.667	0.857	0.750	0.812	0.857
	Random Forest	0.870	16	0	3	4	1.000	0.571	0.727	1.000	0.571
ResNet152	SVM	0.913	16	0	2	5	1.000	0.714	0.833	1.000	0.714
	Gradient Boosting	0.652	12	4	4	3	0.429	0.429	0.429	0.750	0.429
	Logistic Regression	0.870	15	1	2	5	0.833	0.714	0.769	0.937	0.714
	Random Forest	0.783	16	0	5	2	1.000	0.286	0.444	1.000	0.286
VGG16	SVM	0.696	12	4	3	4	0.500	0.571	0.533	0.750	0.571
	Gradient Boosting	0.739	13	3	3	4	0.571	0.571	0.571	0.812	0.571
	Logistic Regression	0.652	12	4	4	3	0.429	0.429	0.429	0.750	0.429
	Random Forest	0.826	14	2	2	5	0.714	0.714	0.714	0.875	0.714
Xception	SVM	0.696	12	4	3	4	0.500	0.571	0.533	0.750	0.571
	Gradient Boosting	0.609	10	6	3	4	0.364	0.571	0.444	0.562	0.571
	Logistic Regression	0.652	12	4	4	3	0.364	0.571	0.445	0.562	0.571
	Random Forest	0.739	11	5	1	6	0.667	0.857	0.750	0.812	0.857

Table 3. Classification approach system with calculated metrics. More values shown in (Annex 3)

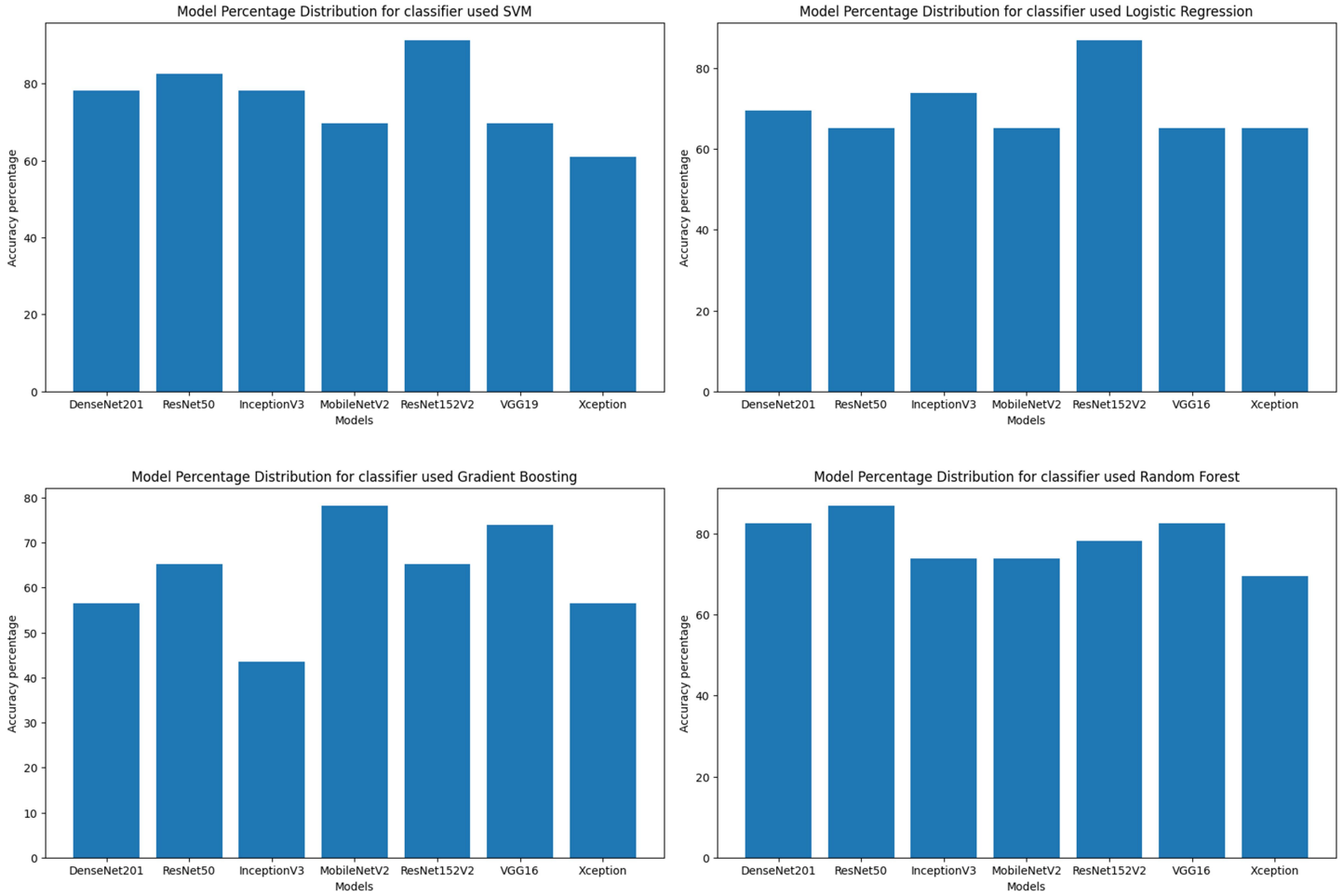


Image 18. Accuracy values classifier performance according to the extracted deep features models

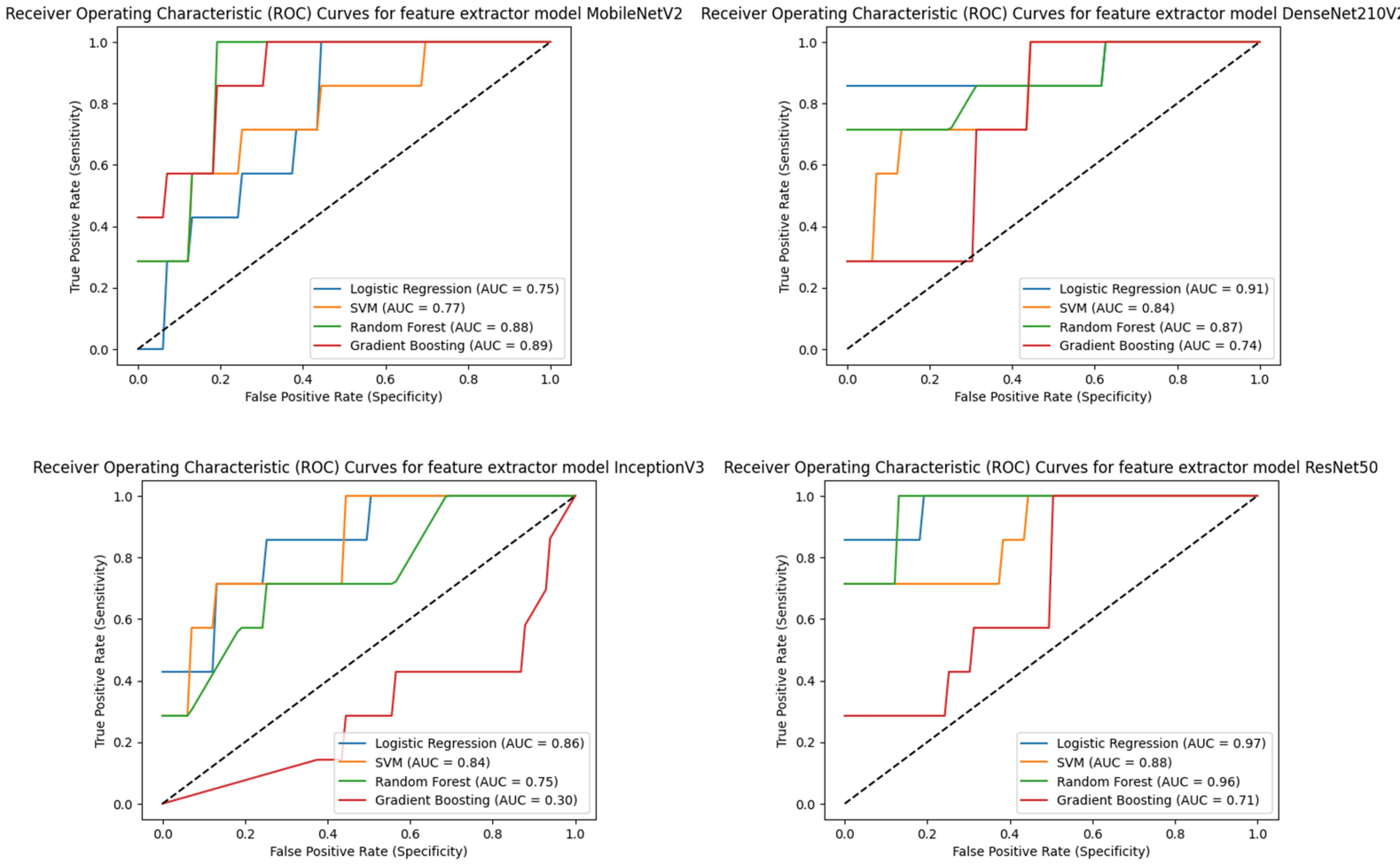


Image 19. ROC curves and AUC values for every classifier model used

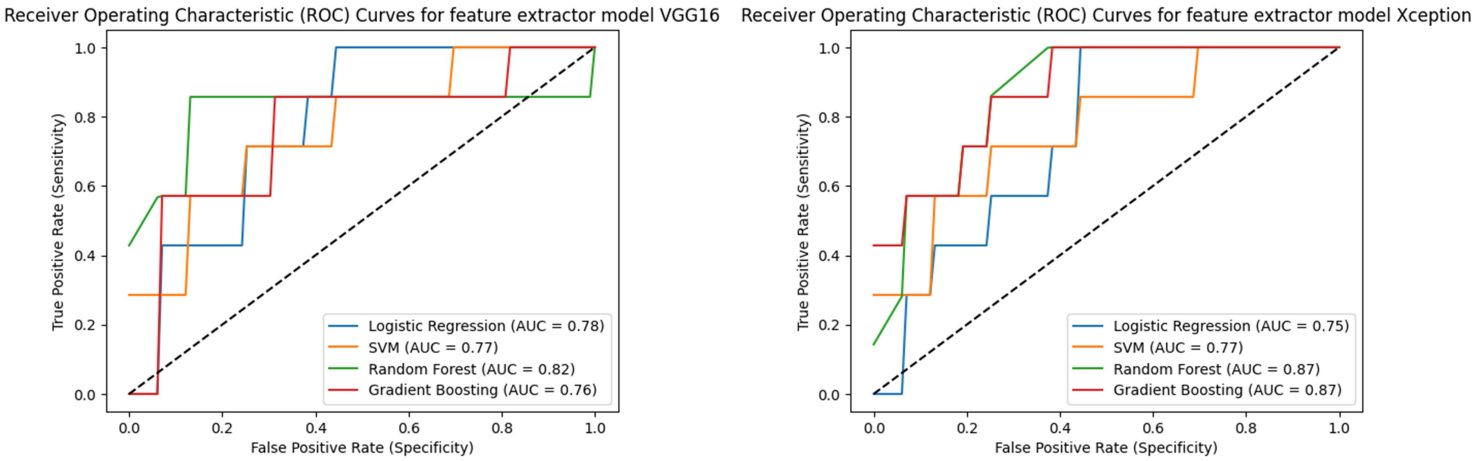


Image 20. ROC curves and AUC values for every classifier model use

	Logistic regression	SVM	Random Forest	Gradient Boosting
MovileNetV2	0,75	0,77	0,88	0,89
DenseNet210V2	0,91	0,84	0,87	0,74
InceptionV3	0,86	0,84	0,75	0,3
ResNet50	0,97	0,88	0,96	0,71
Xception	0,75	0,77	0,81	0,68
VGG16	0,78	0,77	0,82	0,76

Table 4. AUC values for classifiers and extracted features model

The AUC value represents the possibility of the model to distinguish between the negative class and the positive one, in which when the AUC is lower that 0,5 represents a remarkable error in the model. This value would represent that the model performs a totally random prediction, even TP or TN (represented by the dots line in the diagonal of the ROCs curves). In the graphs above we can see that the results are above the line as expected, except for one case the Inceptionv3 model for extraction of deep features + the gradient boosting classifier.

Ideally the ROC curve should look as shown in the following graph:

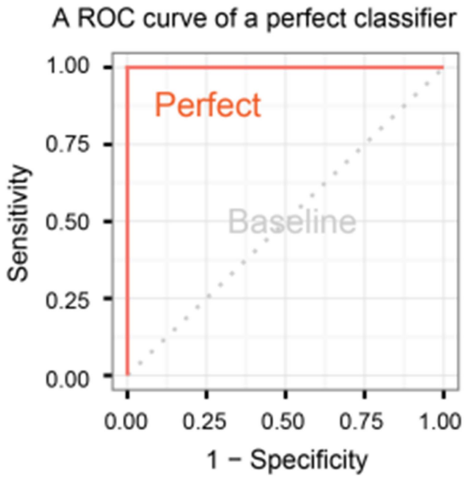


Image 21. Ideal ROC curves

In general the graphics show a good performance respecting to the AUC score, as most of them present values higher than 0.75 in the four classifier models. The

most remarkable value for AUC are accomplished by the Logistic regression classifier model, which achieved 2 methods with higher values of AUC than 0.9.

On the contrary, the worst performance was obtained by the Gradient Boosting classifying method, as it had the lowest values for AUG. In the accuracy measures we can see that it was the weakest in performance as well.

Another remarkable result was that the Random Forest classifier was the more cohesive stable in performance while working with different models to extract features, between all the other classifiers. The AUC value was the least variable between the different deep extracting features models tested. Its accuracy was also the highest and more stable.

For the deep extracting feature models, the best performance through all the classifiers was the DenseNet210V2 and ResNet50 with the highest AUG values; getting the highest one by the ResNet50 working along with the Logistic regression classifier of 0.97. The most stable in the extracting models were the Xception model and the VGG16 model with balanced performances and similar AUC values with different classifiers.

Summing up, we can say that the extracting models worked correctly among with the classifier methods, but for the Gradient Boosting one, which did not showed a good performance. We can say that the ResNet50 was the best model to extract features, as the values obtained were high for every classifier used. This is suggesting the extracted characteristics by this model are useful in this classification. Besides, the worst extractor of features was the InceptionV3 model, as the results obtained were the lowest which referrers that the characteristics obtained by it are not of importance to the target classification.

CONCLUSION

After successfully studied and explored the topic of the project from different angles I accomplished to obtain a system that fulfills the principal objective: a system that detects and classifies polyps. For detection the YoloV7 method was effectively implemented, as it was robustly trained with pictures from various datasets, getting at the end an accuracy value of 0.87. The optimum threshold value was around 0.34 with which images for the classification stage could be obtained greatly sufficiently.

For the classification process GLCMs could be analyzed in two parts to find the best way to extract them and they could correctly merged with other features extract to generate a good prediction of the polyps between Adenomatous ones and hyperplasic. The deep features could be extracted from 6 different models of CNNs, and then merged with the GLCMs, go through 4 different classifier models. Overall, the results were satisfactory. The best performance for the classifier was between the Logistic Regression and the DenseNet210, with the best ROC curves, although the best accuracy was obtained for the Random Forest between all the models for extracting features. Regarding the extracting features, the best results were obtained by the ResNet50 model and the DenseNet210. This remarks that they are extracting the most necessary characteristics from the images for the classification.

For developing this project in the future it would be necessary to confront the biggest problem during the process: the absence of databases on polyps open to the public with classification. For the detection task a lot of information could be found, as a lot about the discoveries already had been made about it and it's a moreover accomplished task, even though is still advancing. In contrary, for classification tasks the proven working systems are yet in growth. Even there are a lot of researchers working on this, and results had been showed, the databases with ground truth available on these images are private information. So, as for the project in the future to work, the idea would be to have the support of a third party, such as medical devices company and/or hospital that could reach a full complete dataset to work with. The system could be proved to be more robust with

more datasets to train and test on, as the dataset was little predicting just a picture wrong or right could change a lot the values for accuracy. Another detail on the matter would be that the possibility to test the system on another database to analyze how accurate it is at the time to change was not possible, as the pictures trained and tested on were mostly from the same investigation. Added to that the project could grow much more by analyzing more characteristics of polyps for the NICE classification, which should be included, as a more in depth color analysis to detect the vessels, other techniques respecting the shape as well.

BIBLIOGRAPHY

[1] Alexander Meining, Ulm; Thomas Rösch, Hamburg. (2019) *Paris Classification: Early Colorectal Cancers.* [Online]. Available: https://www.endoscopy-campus.com/en/classifications/paris-classification-early-colorectal-cancers/

[2] Duan, Y., Jiang, H., Shen, C., & Yu, S. *(2022). DeepFake Detection using Capsule Network with Adversarial Learning. PLOS ONE, 17(4), e0267955.* https://doi.org/10.1371/journal.pone.0267955.

[3] Bernal, J., Histace, Aymeric.*(2021). Computer-Aided Analysis of Gastrointestinal Videos.* Springer.

[4] ASTRE Research Group. (2018). *A Sensor Fault Detection and Identification Algorithm for an Autonomous Surface Vehicle.* HAL. Available: https://hal.laas.fr/ETIS-ASTRE/hal-01896834v1

[5] Cancer Council Australia. (n.d.). *Polyps. Cancer Council Australia.* Available: https://www.cancer.org.au/polyps/

[6] American Cancer Society. (2023). *Key Statistics for Colorectal Cancer. American Cancer Society.* Available: https://www.cancer.org/cancer/types/colon-rectal-cancer/about/key-statistics.html

[7] Mayo Clinic. (2023). *Colon polyps - Symptoms and causes. Mayo Clinic*. Available: https://www.mayoclinic.org/diseases-conditions/colon-polyps/symptoms-causes/syc-20352875

[8] Pimentel-Nunes, P., Dinis-Ribeiro, M., Ponchon, T., Repici, A., Vieth, M., De Ceglie, A., Hassan, C. (2014). *The Paris endoscopic classification of superficial neoplastic lesions: Esophagus, stomach, and colon.* Gastrointestinal Endoscopy, 80(3). https://doi.org/10.1016/j.gie.2014.06.012

[9] National Center for Biotechnology Information. (2018). *Colorectal Cancer: Diagnosis and Clinical Management. NCBI Bookshelf*. Available: https://www.ncbi.nlm.nih.gov/books/NBK430761/

[10] Doubeni C., (2022). *Colon Polyps: Beyond the Basics. UpToDate*. Available: https://www.uptodate.com/contents/colon-polyps-beyond-the-

basics#:~:text=Hyperplastic%20polyps%20%E2%80%94%20Hyperplastic%20polyps%20are,not%20worrisome%20(figure%201).

[11] Cleveland Clinic. (n.d.). *Serrated Polyposis Syndrome/Hyperplastic Polyposis Syndrome*. Cleveland Clinic. Available: https://my.clevelandclinic.org/health/diseases/17462-serrated-polyposis-syndromehyperplastic-polyposis-syndrome#:~:text=Serrated%20polyps%20are%20a%20type,examining%20them%20under%20a%20microscope.

[12] American Cancer Society. (n.d.). *Colon Polyps: Sessile or Traditional Serrated Adenomas. American Cancer Society.* Available: https://www.cancer.org/cancer/diagnosis-staging/tests/understanding-your-pathology-report/colon-pathology/colon-polyps-sessile-or-traditional-serrated-adenomas.html

[13] Iwatate, M., Hirata D., Sano Y., (2020). NBI International Colorectal Endoscopic (NICE) Classification. Springer. Available: https://link.springer.com/chapter/10.1007/978-981-10-6769-3_8#auth-Mineo-Iwatate

[14] Lambin, P., Leijenaar, R. T. H., Deist, T. M., Peerlings, J., de Jong, E. E. C., van Timmeren, J., (2018). *Artificial intelligence in medical imaging: a radiomic guide. Insights into Imaging*, 9(6), 745-762. https://doi.org/10.1007/s13244-018-0639-9.

[15] American Cancer Society. (n.d.). *Colon Polyps: Sessile or Traditional Serrated Adenomas.* Available: https://www.cancer.org/cancer/diagnosis-staging/tests/understanding-your-pathology-report/colon-pathology/colon-polyps-sessile-or-traditional-serrated-adenomas.html.

[16] Siegel R., Sandeep Wagle N., Cercek A., Smith R, Jemal A., (2023), *Colorectal cancer statistics 2023.* Available: https://pubmed.ncbi.nlm.nih.gov/36856579/

[17] American Cancer Society. (n.d.). *Key Statistics for Colorectal Cancer. American Cancer Society*. Available: https://www.cancer.org/cancer/types/colon-rectal-cancer/about/key-statistics.html

[18] Hewett D., Kaltenbach T., Sano Y., Tanaka S., Saunders B., Ponchon T., Soetikno R., Rex D. (2012). *Validation of a Simple Classification System for Endoscopic Diagnosis of Small Colorectal Polyps Using Narrow-Band Imaging.* , 143(3). *doi:10.1053/j.gastro.2012.05.006.*

[19] Endoscopy Campus. (n.d.). *Paris Classification - Early Colorectal Cancers. Endoscopy Campus.* Available: https://www.endoscopy-campus.com/en/classifications/paris-classification-early-cancer/

[20] 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.

[21] Jha, D., Smedsrud, P. H., Riegler, M. A., Halvorsen, P., de Lange, T., Johansen, D., & Johansen, H. D. (2020). *Kvasir-seg: A segmented polyp dataset. In International Conference on Multimedia Modeling* (pp. 451-462). Springer.

[22] Mesejo P., Pizarro D. (2016). *Gastrointestinal Lesions in Regular Colonoscopy. UCI Machine Learning Repository.* https://doi.org/10.24432/C5V02D.

[23] Silva J. S., Histace A., Romain O., Dray X., Granado B. (2014). *Towards embedded detection of polyps in WCE images for early diagnosis of colorectal cancerInternational Journal of Computer Assisted Radiology and Surgery*, Springer Verlag (Germany), 9 (2), pp. 283-293.

[24] Barburiceanu S., Terebes R., Meza S. (2021). 3D Texture Feature Extraction and Classification Using GLCM and LBP-Based Descriptors. 11(5), 2332. *https://doi.org/10.3390/app11052332*

[25] Hall-Beyer, M. (2017). GLCM Texture: A Tutorial (Version 3.0). Retrieved from [URL]. University of Calgary. Available: *https://prism.ucalgary.ca/server/api/core/bitstreams/8f9de234-cc94-401d-b701-f08ceee6cfdf/content*.

[26] Hao Y., Zhang L., Qiao S., Bai Y., Cheng R., Xue H., Hou Y., Zhang W., Zhang G. (2022). *Breast cancer histopathological images classification based on deep semantic features and gray level co-occurrence matrix.* Available: *https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0267955#sec002*.

[27] Yamashita, R., Nishio, M., Do, R.K.G. et al. *Convolutional neural networks: an overview and application in radiology*. Insights Imaging 9, 611–629 (2018). *https://doi.org/10.1007/s13244-018-0639-9*

ANNEX 1

As the GLCM approximates the joint probability distribution of two pixels. A symmetrical matrix around the diagonal means that the same values in pixels occur in cells on opposite sides of the diagonal. Given a normalized GLCM of an image, $P_{i,j}$ is the probability value for the entry cell i,j, being i the row number and j the column number. The equations corresponding to the extracted features are:

$$\text{ASM} = \sum_{i,j=0}^{N-1} {P_{i,j}}^2$$

$$Energy = \sqrt{ASM}$$

When ASM (Angular second moment) and Energy take high values this indicated that the processed windon is ordered.

$$\text{Correlation} = \sum_{i,j=0}^{N-1} P_{i,j} \left[\frac{(i-\mu_i)(j-\mu_i)}{\sqrt{({\sigma_i}^2)({\sigma_j}^2)}}\right]$$

A high value of correlation texture means a high predictability of pixel relationships.

$$\text{Dissimilarity} = \sum_{i,j=0}^{N-1} P_{i,j} \, |i-j|$$

$$\text{Contrast} = \sum_{i,j=0}^{N-1} P_{i,j} \, (i-j)^2$$

A high value of contrast and dissimilarity shows a more contrast in the cell.

$$\text{Homogeneity} = \sum_{i,j=0}^{N-1} \frac{P_{i,j}}{1+(i-j)^2}$$

Homogeneity values are inversed of the contrast weights. A high value of homogeneity indicates a more uniform values distribution in the cell. [25]

ANNEX 2

The accuracy in deep learning models is a basic metric used for model evaluations which describes the number of correct predictions were made by the system over the total predictions.

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

The precision measures how many positive predictions were correct by the system.

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

The Recall or Sensitivity is a value that presents how many positive cases were predicted correctly by the system over all of the positives in the total data.

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

The specificity measures the number of correctly negative predictions that were made.

$$Specificity = \frac{TN}{TN + FP}$$

The F1 score referres to the value of the combination between precision and recall, it describes an average of values.

$$F1 = 2\ \frac{Precision * Recall}{Precision + Recall}$$

ANNEX 3

Models used for extracted deep features	Classifier model	Threshold	Accuracy	Specificity	Sensitivity	TP	TN	FP	FN
DenseNet201	SVM	0,2	0.826	0.938	0.571	15	4	1	3
		0,5	0.783	1.000	0.286	16	2	0	5
		0,7	0.783	1.000	0.286	16	2	0	5
		0,9	0.739	1.000	0.143	16	1	0	6
	Gradient Boosting	0,2	0.565	0.688	0.286	11	2	5	5
		0,5	0.565	0.688	0.286	11	2	5	5
		0,7	0.565	0.688	0.286	11	2	5	5
		0,9	0.609	0.750	0.286	12	2	4	5
	Logistic Regression	0,2	0.696	0.625	0.857	10	6	6	1
		0,5	0.696	0.625	0.857	10	6	6	1
		0,7	0.870	0.875	0.857	14	6	2	1
		0,9	0.957	1.000	0.857	16	6	0	1
	Random Forest	0,2	0.304	0.063	1.000	1	7	15	0
		0,5	0.783	0.875	0.571	14	4	2	3
		0,7	0.696	1.000	0.000	16	0	0	7
		0,9	0.696	1.000	0.000	16	0	0	7
InceptionV3	SVM	0,2	0.739	0.938	0.286	15	2	1	5

		0,5	0.783	1.000	0.286	16	2	0	5
		0,7	0.739	1.000	0.143	16	1	0	6
		0,9	0.739	1.000	0.143	16	1	0	6
	Gradient Boosting	0,2	0.435	0.438	0.429	7	3	9	4
		0,5	0.435	0.438	0.429	7	3	9	4
		0,7	0.435	0.438	0.429	7	3	9	4
		0,9	0.391	0.438	0.286	7	2	9	5
	Logistic Regression	0,2	0.739	0.688	0.857	11	6	5	1
		0,5	0.739	0.750	0.714	12	5	4	2
		0,7	0.826	0.875	0.714	14	5	2	2
		0,9	0.783	0.938	0.429	15	3	1	4
	Random Forest	0,2	0.304	0.000	1.000	0	7	16	0
		0,5	0.739	0.938	0.286	15	2	1	5
		0,7	0.696	1.000	0.000	16	0	0	7
		0,9	0.696	1.000	0.000	16	0	0	7
MobileNetV2	SVM	0,2	0.696	0.875	0.286	14	2	2	5
		0,5	0.739	0.938	0.286	15	2	1	5
		0,7	0.739	1.000	0.143	16	1	0	6
		0,9	0.739	1.000	0.143	16	1	0	6
	Gradient Boosting	0,2	0.739	0.688	0.857	11	6	5	1
		0,5	0.783	0.813	0.714	13	5	3	2
		0,7	0.739	0.813	0.571	13	4	3	3
		0,9	0.739	0.813	0.571	13	4	3	3
	Logistic Regression	0,2	0.652	0.500	1.000	8	7	8	0
		0,5	0.652	0.750	0.429	12	3	4	4

		0,7	0.652	0.750	0.429	12	3	4	4
		0,9	0.739	0.938	0.286	15	2	1	5
	Random Forest	0,2	0.478	0.250	1.000	4	7	12	0
		0,5	0.783	0.813	0.714	13	5	3	2
		0,7	0.739	1.000	0.143	16	1	2	6
		0,9	0.696	1.000	0.000	16	0	0	7
ResNet50	SVM	0,2	0.913	1.000	0.714	16	5	0	2
		0,5	0.783	1.000	0.286	16	2	0	5
		0,7	0.739	1.000	0.143	16	1	0	6
		0,9	0.696	1.000	0.000	16	0	0	7
	Gradient Boosting	0,2	0.609	0.625	0.571	10	4	6	3
		0,5	0.609	0.625	0.571	10	4	6	3
		0,7	0.609	0.625	0.571	10	4	6	3
		0,9	0.652	0.750	0.429	12	3	4	4
	Logistic Regression	0,2	0.870	0.813	1.000	13	7	3	0
		0,5	0.826	0.813	0.857	13	6	3	1
		0,7	0.913	0.938	0.857	15	6	1	1
		0,9	0.913	1.000	0.714	16	5	0	2
	Random Forest	0,2	0.391	0.125	1.000	2	7	14	0
		0,5	0.913	1.000	0.714	16	5	0	2
		0,7	0.696	1.000	0.000	16	0	0	7
		0,9	0.696	1.000	0.000	16	0	0	7
ResNet152	SVM	0,2	0.913	1.000	0.714	16	5	0	2
		0,5	0.739	1.000	0.143	16	1	0	6
		0,7	0.696	1.000	0.000	16	0	0	7

		0,9	0.696	1.000	0.000	16	0	0	7
	Gradient Boosting	0,2	0.652	0.688	0.571	11	4	5	3
		0,5	0.652	0.750	0.429	12	3	4	4
		0,7	0.739	0.875	0.429	14	3	2	4
		0,9	0.783	0.938	0.429	15	3	1	4
	Logistic Regression	0,2	0.870	0.875	0.857	14	6	2	1
		0,5	0.870	0.938	0.714	15	5	1	2
		0,7	0.913	1.000	0.714	16	5	0	2
		0,9	0.870	1.000	0.571	16	4	0	3
	Random Forest	0,2	0.304	0.000	1.000	0	7	16	0
		0,5	0.826	1.000	0.429	16	3	0	4
		0,7	0.696	1.000	0.000	16	0	0	7
		0,9	0.696	1.000	0.000	16	0	0	7
VGG16	SVM	0,2	0.696	0.875	0.286	14	2	2	5
		0,5	0.739	0.938	0.286	15	2	1	5
		0,7	0.739	1.000	0.143	16	1	0	6
		0,9	0.739	1.000	0.143	16	1	0	6
	Gradient Boosting	0,2	0.739	0.813	0.571	13	4	3	3
		0,5	0.739	0.813	0.571	13	4	3	3
		0,7	0.739	0.813	0.571	13	4	3	3
		0,9	0.783	0.875	0.571	14	4	2	3
	Logistic Regression	0,2	0.652	0.500	1.000	8	7	8	0
		0,5	0.652	0.750	0.429	12	3	4	4
		0,7	0.696	0.813	0.429	13	3	3	4
		0,9	0.696	0.938	0.143	15	1	1	6

	Random Forest	0,2	0.478	0.313	0.857	5	6	11	1
		0,5	0.870	0.875	0.857	14	6	2	1
		0,7	0.783	1.000	0.286	16	2	0	5
		0,9	0.696	1.000	0.000	16	0	0	7
Xception	SVM	0,2	0.696	0.875	0.286	14	2	2	5
		0,5	0.739	0.938	0.286	15	2	1	5
		0,7	0.739	1.000	0.143	16	1	0	6
		0,9	0.739	1.000	0.143	16	1	0	6
	Gradient Boosting	0,2	0.565	0.563	0.571	9	4	7	3
		0,5	0.652	0.625	0.714	10	5	6	2
		0,7	0.652	0.688	0.571	11	4	5	3
		0,9	0.696	0.750	0.571	12	4	4	3
	Logistic Regression	0,2	0.652	0.500	1.000	8	7	8	0
		0,5	0.652	0.750	0.429	12	3	4	4
		0,7	0.652	0.750	0.429	12	3	4	4
		0,9	0.739	0.938	0.286	15	2	1	5
	Random Forest	0,2	0.435	0.188	1.000	3	7	13	0
		0,5	0.696	0.688	0.714	11	5	5	2
		0,7	0.783	1.000	0.286	16	2	0	5
		0,9	0.696	1.000	0.000	16	0	0	7

Table 5. More values calculated for different threshold values.