People's Democratic Republic of Algeria

Ministry of Higher Education and Scientific Research

KASDI Merbah University - Ouargla

Faculty of New Information and Communication Technologies

Department of Computing and Information Technology

BY Mohammed el Mehdi KADRI & Abdelmounaim SAGGAI

THEME

Breast Cancer detection using Deep learning

Dr.	Bilal KHALDI.	President	UKM OUARGLA
Dr.	Fouad BEKKARI.	Examiner	UKM OUARGLA
Dr.	Oussama AIADI.	Supervisor	UKM OUARGLA

Academic Year :2021/2022

Acknowledgment

First of all, I would like to thank Allah for all the blessings and favors.

Then I would like to thank my family and friends for the needed support they gave me during my thesis journey, I would like to thank my father for being a source of inspiration to me.

I would like to express my sincere gratitude to our supervisor Dr Oussama AIADI for guiding us through the process of writing this thesis, and helping us in every possible way.

I am also grateful to the jury for honoring us with their presence, and accepting to evaluate our thesis, without forgetting all of my teachers who have helped me during my years in collage, especially Dr AMER Khadija, thank you.

Finally, it wouldn't be an acknowledgment without mentioning my partner who is like a brother to me, the one who never says "no" to anyone, Monaim, thank you for being such a gentil soul.

Kadri Mohammed el mehdi

Acknowledgment

First of all, I would like to present my sincere thankfulness to my dear father and my deceased mother, for their great role in my life.

Then, I would like to express my sincere gratitude to Dr Oussama Aiadi, for his consistent support and guidance during the running of this project.

I am also thankful to the jury for giving us time to evaluate our work.

A special thanks for my dear friend, Yasser, for providing us with the necessary materials in the making of this thesis.

To conclude, I cannot forget to thank yet again my family and friends for all the unconditional support in these very intense academic years.

Saggai Abdelmounaim

Abstract

Breast cancer is among the leading cause of mortality among women in developing as well as under-developing countries. The detection and classification of breast cancer in the early stages of its development may allow patients to have proper treatment. In this article, we proposed some approaches using Artificial intelligent that helps in the detection and classification of breast cancer.

Our first contributions aim to investigate some different CNN architectures for breast cancer.

Then we tried a different approach using DCTNet architecture, the process is faster compared to CNN.

In the last contribution we tried to investigate the fusion of the last two methods, to see if there would be any improvements.

Keywords: Breast cancer, Artificial intelligent, CNN, DCTNet.

Résume

Le cancer du sein est l'une des principales causes de mortalité chez les femmes dans les pays en développement et sous-développés. La détection et la classification du cancer du sein dans les premiers stades de son développement peuvent permettre aux patientes d'avoir un traitement approprié. Dans cet article, nous avons proposé quelques approches utilisant l'intelligence artificielle qui aident à la détection et à la classification du cancer du sein. Notre contribution vise à étudier différentes architectures de CNN qui concerne le cancer du sein.

Ensuite, nous avons essayé une approche différente en utilisant l'architecture DCTNet, le processus est plus rapide par rapport à CNN.

Dans la dernière contribution, nous avons essayé d'étudier la fusion des deux dernières méthodes, pour voir s'il y aurait des améliorations.

Mot clé : Cancer du sein, l'intelligence artificielle, CNN, DCTNet.

ملخص

يعد سرطان الثدي من بين الأسباب الرئيسية للوفيات بين النساء في معضم البلدان. قد يتيح الكشف عن سرطان الثدي وتصنيفه في المراحل الأولى للحصول على العلاج المناسب في اقرب وقت ممكن. في هذه المقالة اقترحنا بعض الأساليب باستخدام الذكاء الاصطناعي الذي يساعد في الكشف عن سرطان الثدي وتصنيفه. تهدف الأولى إلى التحقيق في بعض هياكل CNN المختلفة لسرطان الثدي.

ثم جربنا طريقة مختلفة باستخدام بنية DCTNet ، فالعملية أسرع مقارنة بـ CNN.

في المساهمة الأخيرة ، حاولنا التحقق من اندماج الطريقتين الأخيرتين ، لمعرفة ما إذا كان سيكون هناك أي تحسينات.

الكلمات المفتاحية : سرطان الثدي , CNN, DCTNet, الذكاء الاصطناعي,

Table of Contents

List of figures

List of tables

1. General introduction:

The development of some technologies made such a big impact on the commune world of science. Radiology in particular is one of the most revolutionary innovations in the medical world.

The use of ultrasound in medicine goes back to 18^{th} and 19^{th} centuries, it was made possible by the development of the use of sound in the mid-20^{th}, and came the computer and changed everything.

Breast cancer, is the most commune cancer with 7.4 million cases yearly, 2.4 million dies, making it the most spread cancer.

The early detection of Breast cancer is very crucial, it could prevent many upcoming scenarios that could lead to probably death. The regular most commune way to diagnose Breast cancer is by the diagnosis of a pathologist. Even if it’s the most efficient way, it has some disadvantages like: (1) professional background and rich experience of pathologists are so difficult to inherit or innovate that primary-level hospitals and clinics suffer from the absence of skilled pathologists, (2) the tedious task is expensive and time-consuming, and (3) over fatigue of pathologists might lead to misdiagnosis. It is extremely urgent and important to use computer-aided breast cancer multi-classification, which can reduce the heavy workloads of pathologists and help avoid misdiagnosis.

1.1. Problematic:

In 2020, That were 685 000 deaths caused by Breast cancer, Breast cancer appears in every country in the world in woman of any age after puberty but with increase of the risk in later life.

Recently, many applications using deep learning in the detection of breast cancer are still in progress, However, the use of these applications isn’t 100% efficient.

1.1 Motivations:

Regarding the massive improvement; AI has given the medical field over the past few years, we were motivated by the potential of the use of Machine Learning in the early detection and classification of the breast cancer, and the huge impact it has on overcoming the burdens and difficulties existing in the medical field.

1.2 Contributions:

We proposed a convolutional neural network (CNN), by investigating different deep CNNs for breast cancer, then we tried a different approach by using an unsupervised DCTNet architecture which computationally faster than the CNN, after that we inspected the combination of them both.

We aim to optimize or increase the accuracy in the use of DL for identifying and classifying Breast cancer.

We carried out our work with extensive experiments to assess the performance of our proposed methods.

2. Chapter 1: Work-Background

2.1 Introduction:

The use of artificial intelligence (AI) in diagnostic medical imaging is undergoing extensive evaluation. AI has shown impressive accuracy and sensitivity in the identification of imaging abnormalities and promises to enhance tissue-based detection and characterization, Recently, deep learning-based AI techniques have been actively investigated in medical imaging, in this review, we introduce the main approaches we started our work from (CNN-Digital imaging-DCTNet…) and the main components we based our work on.

2.2 Artificial intelligent:

Artificial intelligence (AI), the ability of a digital computer or machine to perform tasks commonly performed by humans.

2.3 Machine Learning:

Is a part of artificial intelligence, is the study of computer algorithms that can improve automatically through experience and by the usage of some data. Machine learning algorithms build a model based on sample data, known as training data, in order to make predictions or decisions without being explicitly programmed to do so. Machine learning algorithms are used in a wide variety of applications, such as in medicine, email filtering, speech recognition, and computer vision.

2.3.1 Approches :

Machine learning approaches are traditionally divided into four broad categories:

A. Supervised Learning:

Supervised learning algorithms build a mathematical model of a set of data that contains both the inputs and the desired outputs. The data is known as training data, and consists of a set of training examples. Each training example has one or more inputs and the desired output, also known as a supervisory signal.

Examples of supervised learning algorithms:

- SVM (Support Vector Machine)
- Random forest

B. Unsupervised Learning:

Unsupervised learning algorithms take a set of data that contains only inputs, and find structure in the data, The algorithms, therefore, learn from test data that has not been labeled, classified or categorized. Instead of responding to feedback.

Examples of unsupervised learning algorithms:

- KNN (The k-nearest neighbors algorithm).
- K-means clustering.

C. Semi-supervised Learning:

Semi-supervised learning falls between unsupervised learning (without any labeled training data) and supervised learning (with completely labeled training data). Some of the training examples are missing training labels.

D. Reinforcement Learning:

Reinforcement learning is the learning of a mapping from situations to actions so as to maximize a scalar reward or reinforcement signal. The learner is not told which action to take, as in most forms of machine learning, but instead must discover which actions yield the highest reward by trying them

2.4 Pattern Recognition:

Pattern recognition the usage of ML algorithms to recognize patterns. Pattern recognition can be defined as the classification of data based on knowledge already acquired. One of the important aspects of pattern recognition is it potential in applications.

Examples: Speech recognition, speaker identification, multimedia document recognition (MDR), automatic medical diagnosis.

Pattern recognition possesses the following features:

- Pattern recognition system should recognize familiar patterns quickly and accurate.
- Recognize and classify unfamiliar objects.
- Accurately recognize shapes and objects from different angles.
- Identify patterns and objects even when partly hidden.
- Recognize patterns quickly with ease, and with automaticity.

2.5 Digital Imaging:

A digital image is an image composed of pixels, each with finite, discrete quantities of numeric representation for its intensity or gray level.

Digital image has two types

- Vector
- Raster

A. Pixel :

A pixel is the smallest unit of a digital image or graphic that can be displayed and represented on a digital display device.

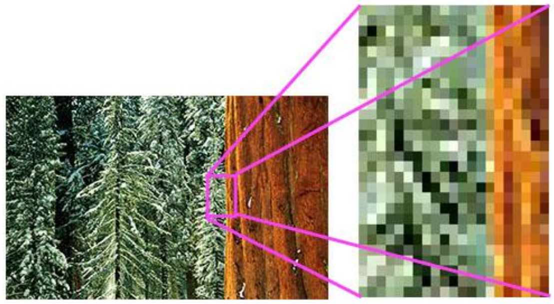

Figure 1 Pixel

B. Characteristics:

The General characteristics of any image are:

A. Résolution :

Resolution refers to the smallest size an object or detail can be represented in an image. Higher resolution means that pixel sizes are smaller, providing more detail.

B. Contrast:

Is the relation between the darkest and brightest parts of an image. If the difference between darkest and lightest portions of an image is vast it is said to have high contrast. On the other hand, if an image's tonal range is more toward grayer tones and the shadows are not very dark while highlights are not very bright, an image has low contrast.

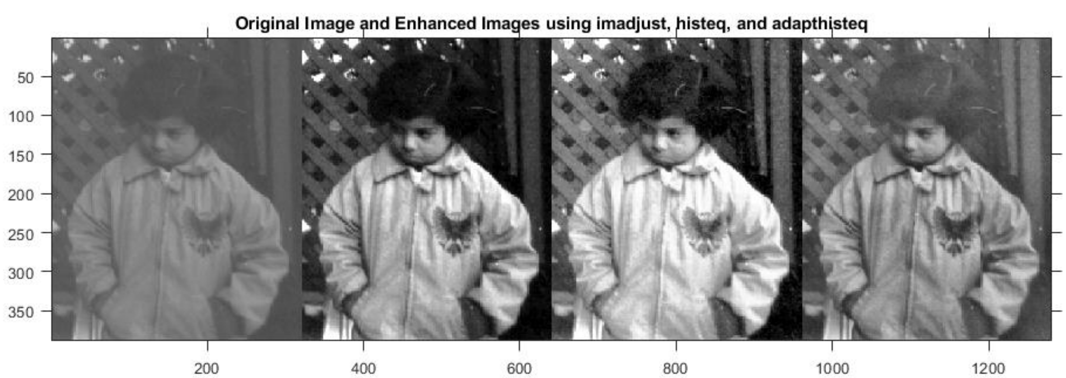

Figure 2 Contrast

C. Dimension:

The size of an image can also be described as its dimension. Usually, it is given in pixels in the format width x height.

D. Histogram :

An image histogram is a graphical representation of the number of pixels in an image as a function of their intensity.

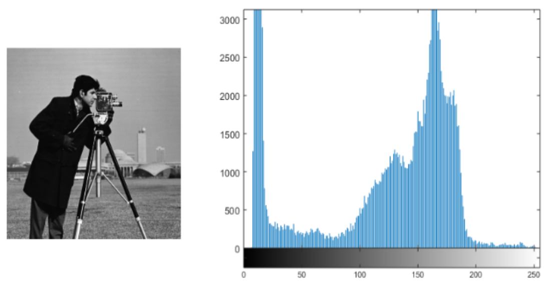

Figure 3 Histogram

E. Color image:

A color image is a picture displayed in color by a computerized device on an attached or separate display screen.

2.6 Medical Imaging:

Medical imaging remains one of the best ways to diagnose patients, as it allows us to see what's going on inside the body without the need for surgery or other invasive procedures. And it refers to several different technologies that are used to view the human body in order to diagnose, monitor, or treat medical conditions. Each type of technology gives different information about the area of the body being studied or treated, related to possible disease, injury, or the effectiveness of medical treatment (1).

2.6.1 Types of Medical Imaging:

There is several types of medical imaging technologies:

A. Computed Tomography:

Computed tomography, often referred to as CT or CAT scanning (Computerized Axial Tomography), is a medical imaging technology that uses X-ray radiation. Images are created when X-rays pass through a patient's body and specialized detectors capture the exiting X-rays, converting this information to a visible image. Computed tomography scans take several images through continuous sections of a patient's body or body part (1). This creates a set

of cross-sectional images that provide information about bones, tissues, and blood vessels.

Computed tomography scans can be more effective than plain X-rays because they are more detailed, but they do require higher doses of radiation. Doctors often use CT scans to diagnose internal injuries after an accident, locate a tumor, or detect a disease such as cancer (1).

Figure 4 Breast cancer (Chest X-ray)

B. Magnetic Resonance imaging:

Magnetic resonance imaging (MRI) utilizes superconducting magnets and radio waves to form images rather than ionizing radiation. An MRI machine consists of a large magnet that creates a magnetic field. An MRI scan uses a strong magnetic field and radio waves to generate images of organs and tissues. Doctors choose to use MRI when they want to analyze a patient's ligaments and tendons, soft tissues, or organs (1) (2). MRI of the brain can help doctors diagnose strokes, tumors, eye disorders, aneurysms, and other conditions. Magnetic resonance imaging can also help doctors detect tumors or cancer in a patient's liver, breast, ovaries, kidney, pancreas, and other organs (1).

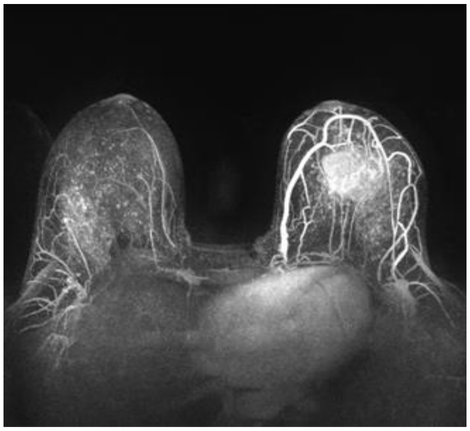

Figure 5 An MRI of a breast tumor

C. Ultrasound scan:

An ultrasound scan, sometimes called a sonogram, is a procedure that uses high-frequency sound waves to create an image of part of the inside of the body.

An ultrasound scan can be used to monitor an unborn baby, diagnose a condition, or guide a surgeon during certain procedures.

Our Database is only based on this one particular scan.

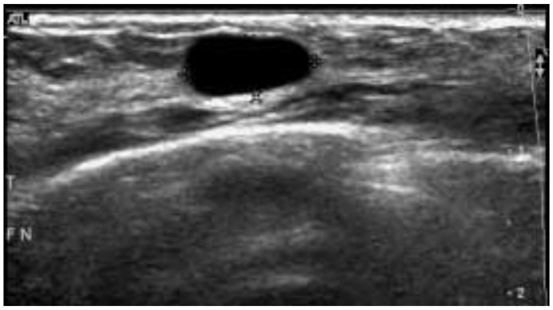

Figure 6 An Ultrasound of a tumor

2.7 Deep Learning

Deep learning is a subset class of machine learning, which is essentially a neural network with three or more layers. These neural networks attempt to simulate the behavior of the human brain far from matching its ability, allowing it to “learn” from large amounts of data (2). (3)

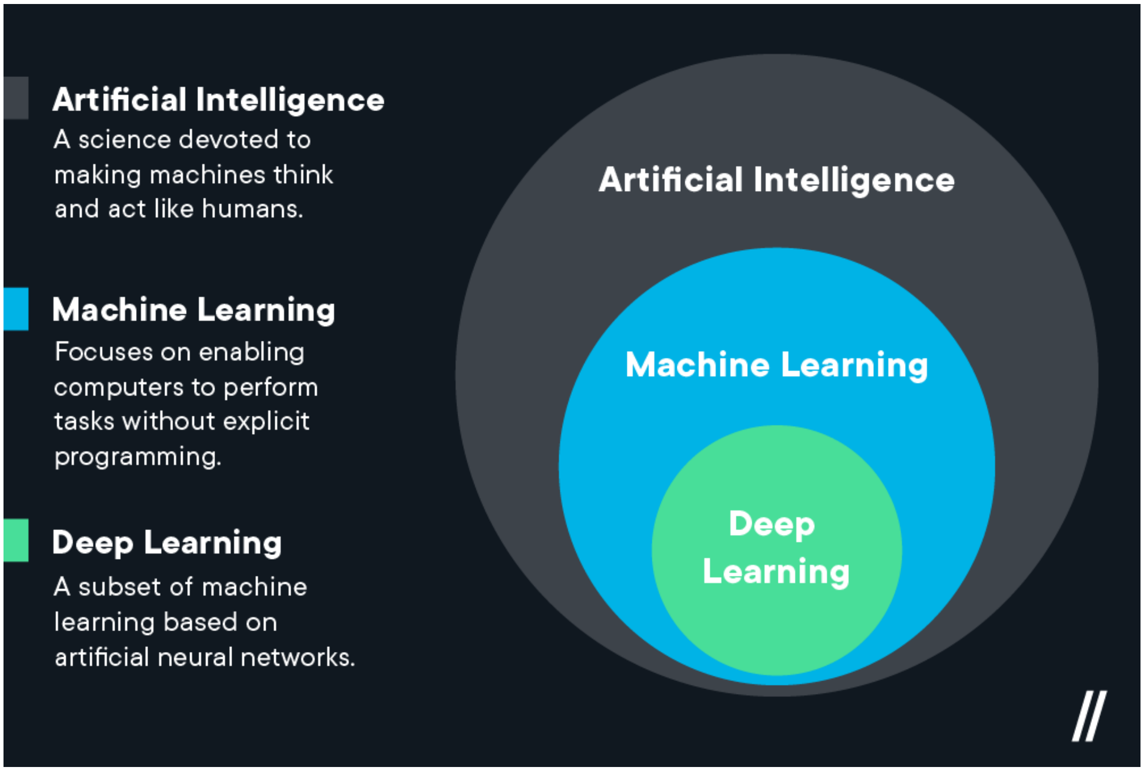

Figure 7Relationship between the AI and ML and DL

2.8 Artificial neural network (ANN):

Neural networks, also known as artificial neural networks (ANNs) are a subset class of machine learning and the main unit of deep learning algorithms. Their structure is inspired by the human brain, copying the way that biological neurons signal to one another.

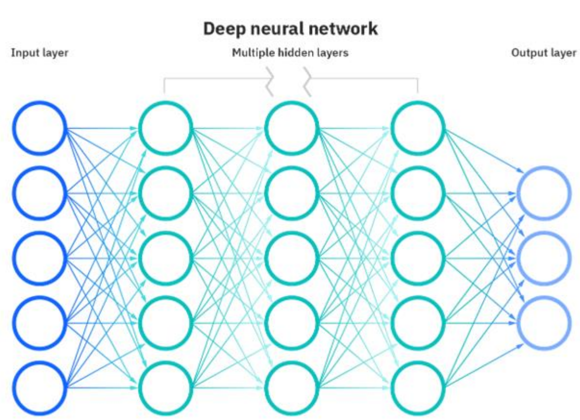

Figure 8 Neural network.

2.9 Convolutional neural network (CNN):

A convolutional neural network, or CNN, is a deep learning neural network designed for processing structured arrays of data such as images.

2.9.1 CNN Architecture:

The Convolutional neural network architecture contains two main parts:

- A convolution tool that identifies and separate the various features of the image for analysis in a process called Feature extraction.
- A fully connected layers that utilize the output from the convolutional process and predicts the class of the image based on the features extracted in the previous stages.

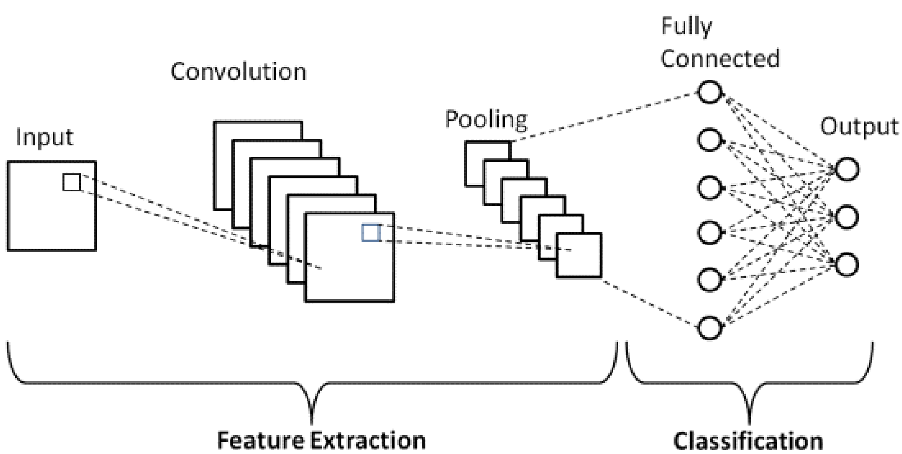

Figure 9 CNN Architecture

Here’s some of the most popular CNNs:

- Xception model
- VGG model
- ResNet model
- Inception model
- Inception Resnet
- MobileNet model
- DenseNet model

2.10 The Discrete Cosine Transform Network (DCTNet):

The Discrete Cosine Transform (DCT) takes data in one form and convert it to another form such that the output data is arranged in decreasing order of importance, this allows us to use only a few of the output data points and still get back to original data (3) (4).

The DCTNet is a process that allow us to apply a classification algorithm (SVM, KNN…) on a Data that was converted to another form (histogram vector) by the DCT.

2.11 Conclusion:

In this chapter, we introduced the different basic fundamentals we should know to start our work on the detection and classification of breast cancer using deep learning.

We started by getting to know what is an artificial intelligence, and defining image and Digital image, then explaining the term medical imaging and its different modalities. we presented a sub class of AI called Machine learning and the deference between the ML and the deep learning. After that we dived into the main methods of machine learning (ANN-CNN), then in the last section we introduced a technique called DCTNet.

In the next chapter we will focus more on the different Approaches that we proposed for the detection and classification of breast cancer.

3. Chapter 2: Proposed methods

3.1 Introduction:

As we have seen in the previous chapter one of the most efficient methods used in the AI field is The CNN which is part of the computer vision, which it may cost some heavy resources in it training, while the DCTNet method requires less and faster in training then the CNN.

For this purpose, we tried to use them both separately to analyze the results for each one, then we tried to combine then to see if there would be an improvement in the results.

3.2 Proposed methods:

3.2.1 Contribution 1: The use of CNN:

At first, we used popular CNNs and a small number of epochs, then we tried to add some layers or freeze the trans on some, and we tried to fuse some CNNs together.

The output is displayed in a vector of probabilities of each class (“we used SoftMax activation in the last layer”).

A. Activation function used:

Rectified Linear unit (ReLU):

in this function, outputs for the positive inputs can range from 0 to infinity but when the input is zero or a negative value, the function outputs zero and it hinders with the back-propagation. This problem is known as the *dying ReLU problem.*

Mathematically it can be represented as:

Equation 1 ReLU

$$f(x) = max(0, x)$$

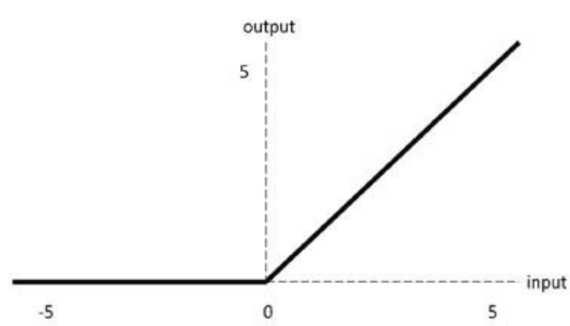

Figure 10 ReLU function

SoftMax function:

the SoftMax function is described as a combination of multiple sigmoids.

It calculates the relative probabilities. Similar to the sigmoid/logistic activation function, the SoftMax function returns the probability of each class.

It is most commonly used as an activation function for the last layer of the neural network in the case of multi-class classification.

Mathematically it can be represented as:

Equation 2 SoftMax

$$SoftMax(z_{i)} = \frac{exp(z_i)}{\sum_i exp(z_i)}$$

B. Loss function used:

Categorical cross-entropy:

It is a loss function that is used for single label categorization. This is when only one category is applicable for each data point. In other words, an example can belong to one class only.

Equation 3 Categorical cross-entropy

$$L(y, \hat{y}) = -\sum_{j=0}^{M}\sum_{i=0}^{N}(y_{ij*\log(\hat{y}_{ij})})$$

Y: Represents the True value.

Ŷ: Represents the Predict value.

C. Layers:

As we said in the previous chapter the CNN Architecture contains two main parts each part has many layers.

Feature extraction:

Feature extraction is the process of highlighting a group of features, which surely represent the information that is important for analysis and classification.

Convolutional layer:

This layer is the first layer that is used to extract the features from the input image. In this layer, the operation is performed between the input image and a filter of a particular size. The output is termed as the Feature map which gives us information about the image. Later, this feature map is given to other layers.

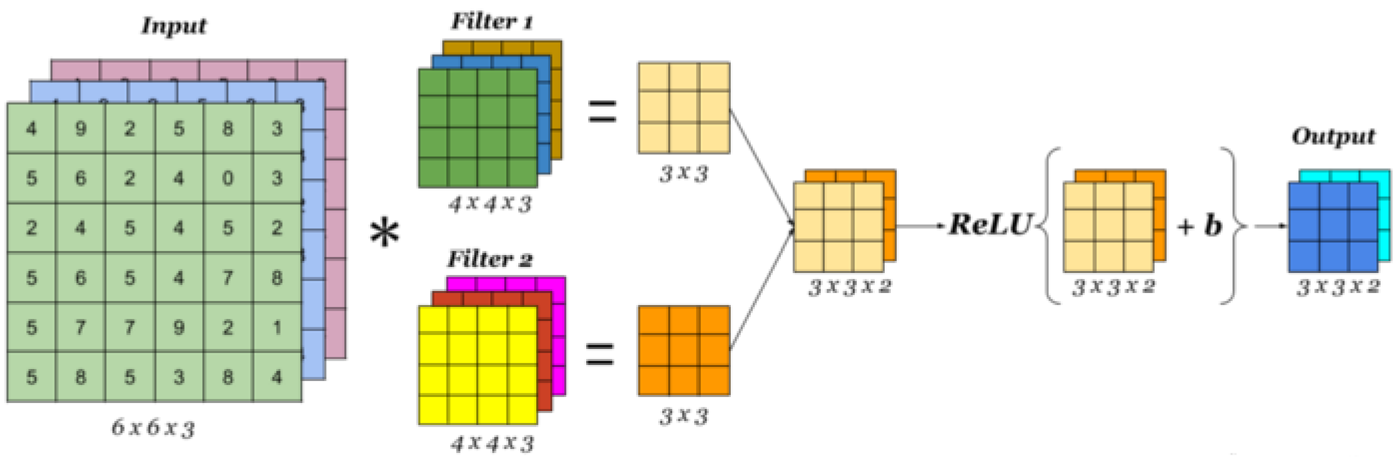

Figure 11 Convolutional layer

Pooling layer:

The primary purpose of this layer is to decrease the size of the obtained feature map to reduce the costs. Depending on the method used, there are several types of Pooling operations.

a) Max Pooling layer:

In max pooling, from each patch of a feature map, the maximum value is selected to create a reduced map.

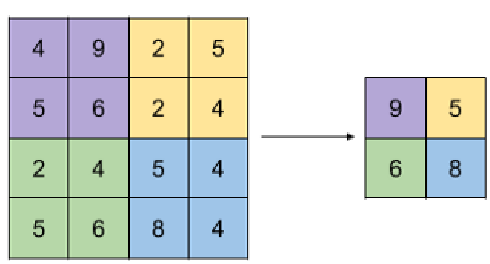

Figure 12 Max Pooling

b) Average Pooling layer:

In average pooling, from each patch of a feature map, the average value is selected to create a reduced map.

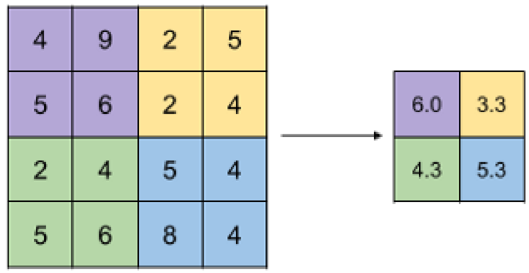

Figure 13 Average Pooling

c) Min Pooling layer:

In min pooling, from each patch of a feature map, the minimum value is selected to create a reduced map.

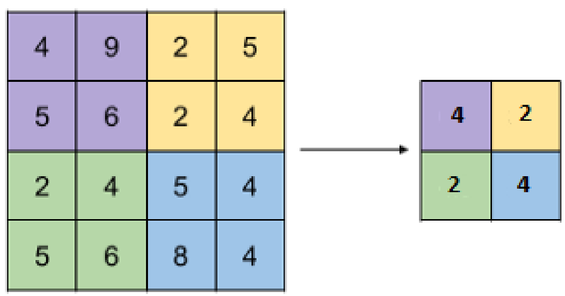

Figure 14 Min Pooling

Classification:

a) Flatten Layer:

Flattening is used to convert all the resultant 2-Dimensional arrays from pooled feature maps into a single long continuous linear vector.

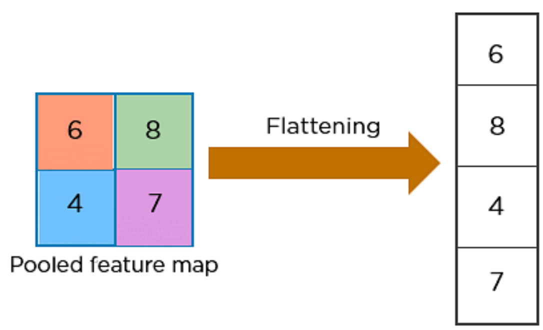

Figure 15 flattening Process

b) Dense layer:

Dense Layer is simple layer of neurons in which each neuron receives input from all the neurons of previous layer, thus called as dense. Dense Layer is used to classify image based on output from convolutional layers. Working of single neuron. A layer contains multiple number of such neurons.

c) Fully connected layers:

Fully Connected layers in a neural network are those layers where all the inputs from one layer are connected to every activation unit of the next layer.

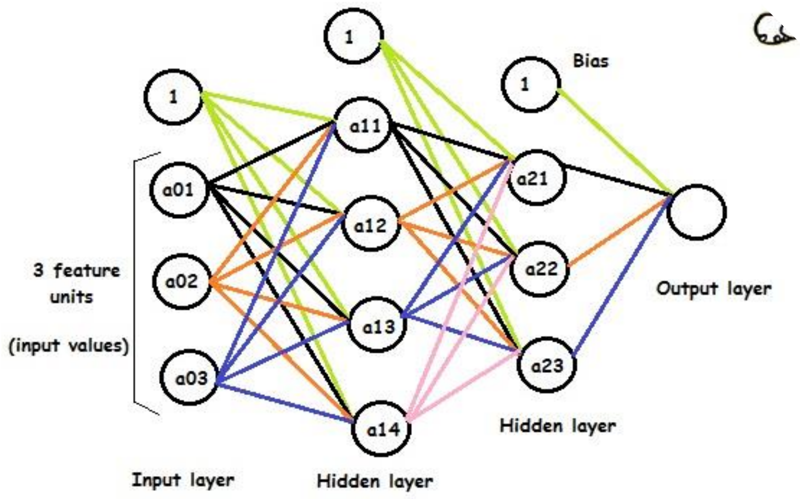

Figure 16 A Neural Network with Fully connected Layers

d) Output Layer:

The output layer in an artificial neural network is the last layer of neurons that produces given outputs for the program.

Other layers:

There are some other layers that could be found in the feature extraction part or in the classification, here the ones we used:

Batch normalization Layer:

Batch normalization is a technique for training very deep neural networks that standardizes the inputs to a layer for each mini-batch.

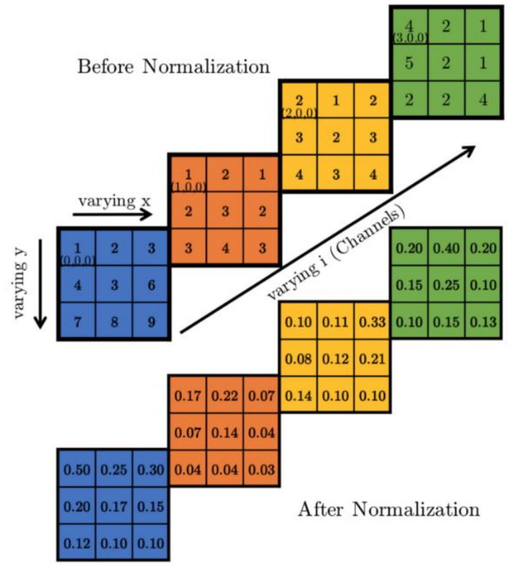

Figure 17 Batch Normalization Example

Dropout Layer:

Dropout is a technique that drops neurons from the neural network or 'ignores' them during training.

D. The CNNs we used:

We used many CNN architectures, some popular ones, and some we created on our own.

Like the ResNet architecture which is, an artificial neural network (ANN). It is a gateless or open-gated variant of the Highway Net, the first working very deep feedforward neural network with hundreds of layers, much deeper than previous neural networks (4) (5).

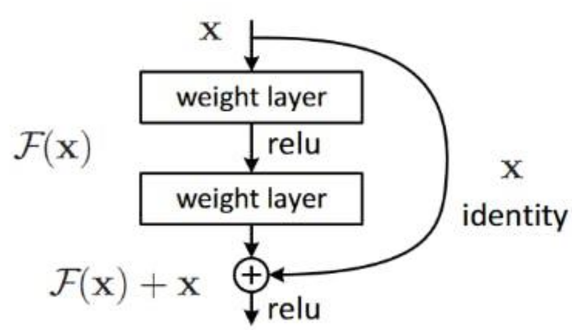

Figure 18 ResNet Architecture

There are many forms of ResNet that we used, like ResNet50, ResNet50V2, ResNet152, the deference between each one is the number and the complexity of blocks.

Also, we used a VGG model that is, is an innovative object-recognition model that supports up to 19 layers.

We used VGG19, and VGG16 in our work, The concept of the VGG19 model (also VGGNet-19) is the same as the VGG16 except that it supports 19 layers. The "16" and "19" stand for the number of weight layers in the model (convolutional layers). This means that VGG19 has three more convolutional layers than VGG16.

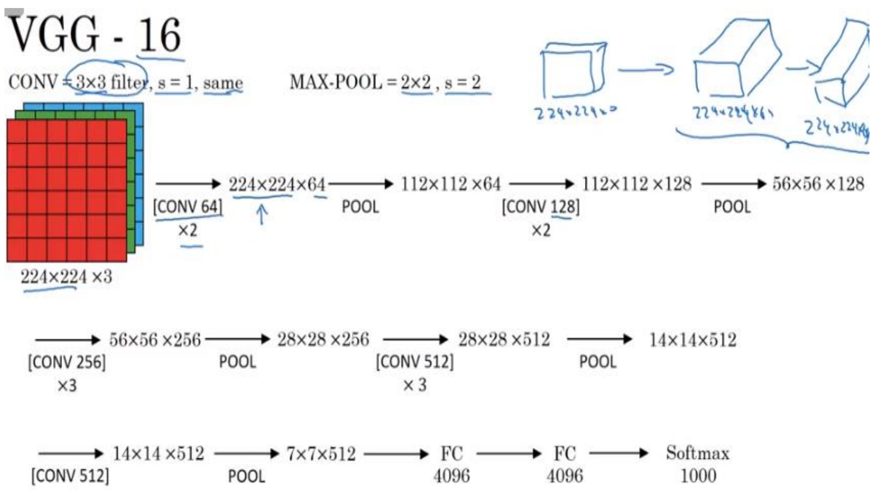

Figure 19 VGG-16 Architecture

And we tried the Xception model which is a convolutional neural network that is 71 layers deep.

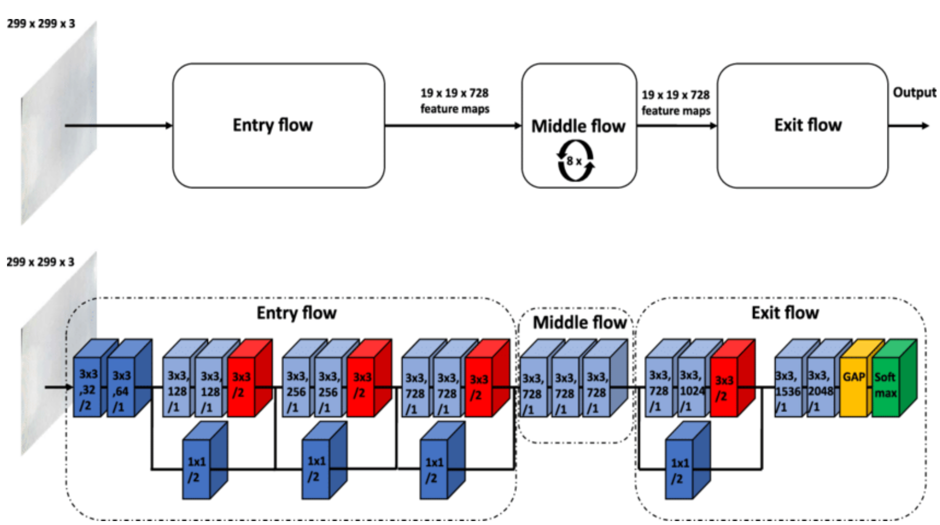

Figure 20 4 Xception model Architecture

Then we worked with the inception model, which is an image model block that aims to approximate an optimal local sparse structure in a CNN.

We only used the one form of the inception model (Inception V3), V3 stands version 3.

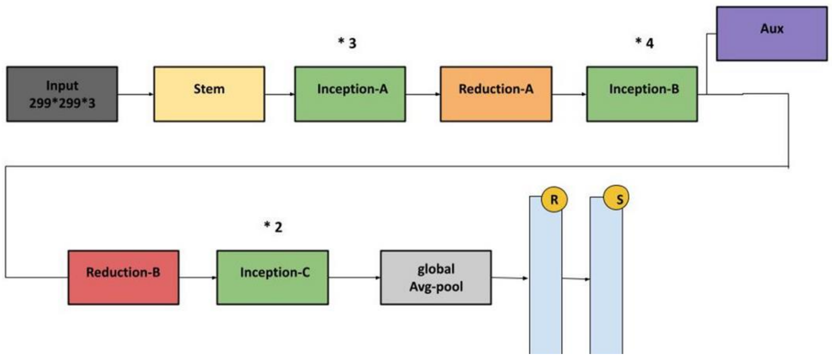

Figure 21 Inception V3 explained

We found a model that combine two models, InceptionResNetV2 which is a convolutional architecture that builds on the inception family of architectures but incorporates residual connections.

Using MobileNet which is an efficient and portable CNN architecture that is used in real world applications. We used two forms (MobileNet-V1, MobileNet-V2) the difference between the two is in the complexity.

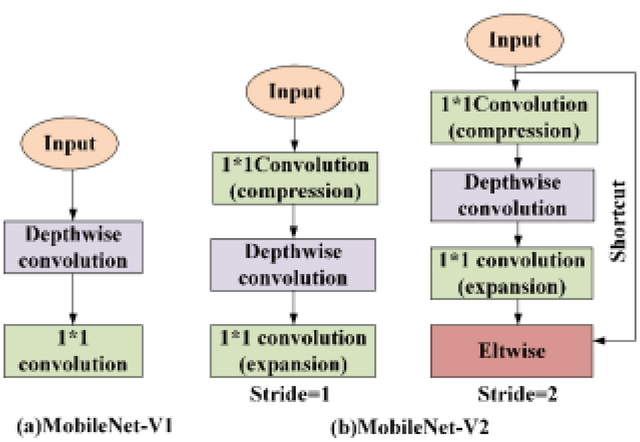

Figure 22 The structural module of MobileNet-V1 and MobileNet-V2

And we tried with the DenseNet model, which is a type of convolutional neural network that utilizes dense connections between layers.

We used three forms (DenseNet121, DenseNet169, DenseNet201) the difference between them all is in the number of the layer blocks.

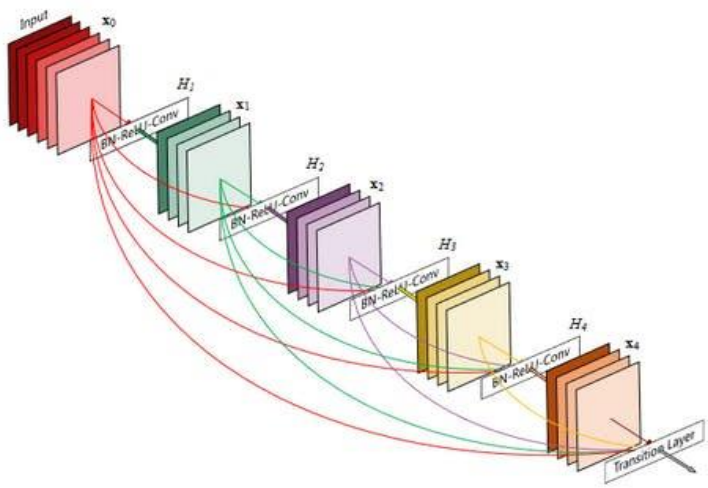

Figure 23 DenseNet Architecture

EfficientNet is a convolutional neural network architecture and scaling method. We used many forms (EfficientNetB0, EfficientNetV2B0, EfficientNetB1, EfficientNetV2B1, EfficientNetB2, EfficientNetV2B2, EfficientNetB3, EfficientNetV2B3, EfficientNetB5, EfficientNetB7) The deference between them is in the number of layer blocks or the complexity.

And we created some models, by using some layers or using the famous CNNs in another way (either by adding some block layers to them, or freezing some other).

Most of these models we tried our own architecture, only the model 6 we took it from Kaggle where it was originally used for bone cancer, the results were unpredictable, we will dive into them in the next chapter.

3.2.2 Contribution 2: <u>The use of DCTNet</u>:

Here we generated 64 filters the size of each one is 8x8, we picked random filters because we couldn't test them all and it would take too much time, we were picking 3 filters at the time then we tried with 7 filters, then we used the DCTNet regular steps, also we used some algorithms of classification.

The output is a vector of probabilities of each class ("we used the predict (predict_poll method)").

A. DCTNet Steps:

Step1: Generating the DCT filter:

The DCT got a fixed number of filters depending on the size of each one, the filters are generated by a mathematical equation:

Equation 4 equation to generate filters

$$\cos\left[\frac{(2x+1)u\pi}{16}\right]\cos\left[\frac{(2y+1)v\pi}{16}\right]$$

x: pixel index column

y: pixel index row

u: The horizontal spatial frequency

v: the vertical spatial frequency

Step2: Convolutional layers:

We use the generated filter in a convolutional operation with the input image to create a feature map for each filter.

Step3: Binarization:

The Binarization is performed on the feature map by binarizing each value (value one if its positive, zero if its negative), whenever we find a value of one means that the filter responded in this particular pixel.

Step4: Binary-Hashing:

We take all the binarized feature maps and we put each number from the same pixel in a new pixel together respectively, then we change it value to decimal.

Step5: Blockwise histograming:

We divide the last output from the binary hashing into a number of blocks with a certain size, then we create a histogram for each block, after that we concatenate all the histograms together.

Step6: Classification:

After finishing all the images in the Dataset, we apply a classification algorithm on the outputs of the blockwise histograming, some of the algorithms that we used:

a) The K-Nearest neighbors Algorithm (KNN):

The k-nearest neighbors algorithm, also known as KNN or k-NN, is a non-parametric, supervised learning classifier, which uses proximity to make classifications or predictions about the grouping of an individual data point (5).

To calculate the distance between the neighbors we used two Distance functions:

1.Euclidean distance function:

Equation 5 Euclidean Distance function.

$$d(x_i, x_j) = \sqrt{\sum_{k=1}^{p} (x_{ik} - x_{jk})^2}$$

d: Represents the distance calculated.

P: Represents the number of dimensions.

x_i, x_j : Two points where the distance is calculated.

2.Cosine distance function:

Equation 6 Cosine Distance function.

$$d(x_i, x_j) = \frac{\sum_{k=1}^{p} x_{ik} x_{jk}}{\sqrt{\sum_{k=1}^{p} {x_{ik}}^2} \sqrt{\sum_{k=1}^{p} {x_{jk}}^2}}$$

b) Logistic regression algorithm:

This type of statistical model (also known as logit model) is often used for classification and predictive analytics. Logistic regression estimates the probability of an event occurring based on a given dataset of independent variables. The dependent variable is bounded between 0 and 1.

This algorithm uses this function to predict:

Equation 7 Equation of the prediction used by the logistic regression algorithm

$$\hat{Y} = \frac{1}{1 + exp[-(\hat{b} + \hat{w}_1 . x_1 + \ldots \hat{w}_n . x_n)]}$$

Ŷ: Represents the Predict value.

$\hat{b}$: Represents the Biase.

$\hat{w}$: Represents the weight.

x_1: Represents a feature.

c) Naïve Bayes algorithm:

The Naive Bayes classification algorithm is a probabilistic classifier. It is based on probability models that incorporate strong independence assumptions.

We used two types of naïve bayes:

1.Gaussian.

2.Naive bayes multinomial.

d) Decision Tree algorithm:

Decision Trees are a type of Supervised Machine Learning where the data is continuously split according to a certain parameter.

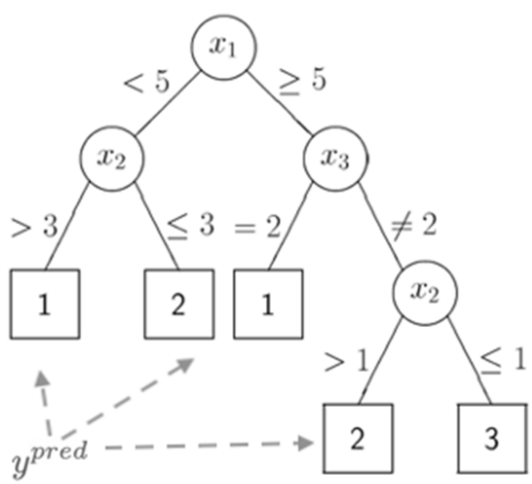

Figure 28 Decision tree example

e) Random Forest algorithm:

Random forest is a Supervised Machine Learning Algorithm that is used widely in Classification problems. It builds decision trees on different samples and takes their majority vote.

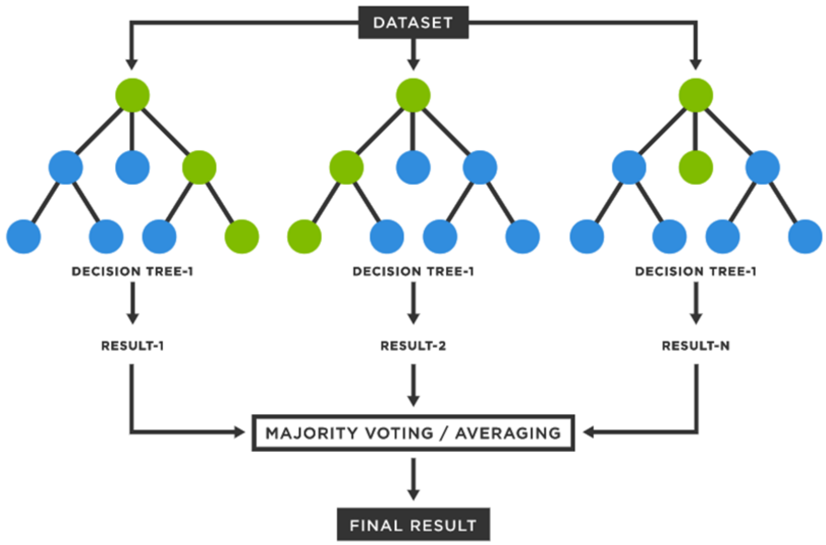

Figure 29 Random forest Processing.

f) Gradient Boosting algorithm:

Gradient boosting relies on the intuition that the best possible next model, when combined with previous models, minimizes the overall prediction error.

g) stochastic gradient descent algorithm:

Stochastic gradient descent is an optimization algorithm often used to find the model parameters that correspond to the best fit between predicted and actual outputs.

d) support vector machine algorithm

Support Vector Machine (SVM) is a supervised machine learning algorithm used for classification. We worked with a support vector classifier (SVC).

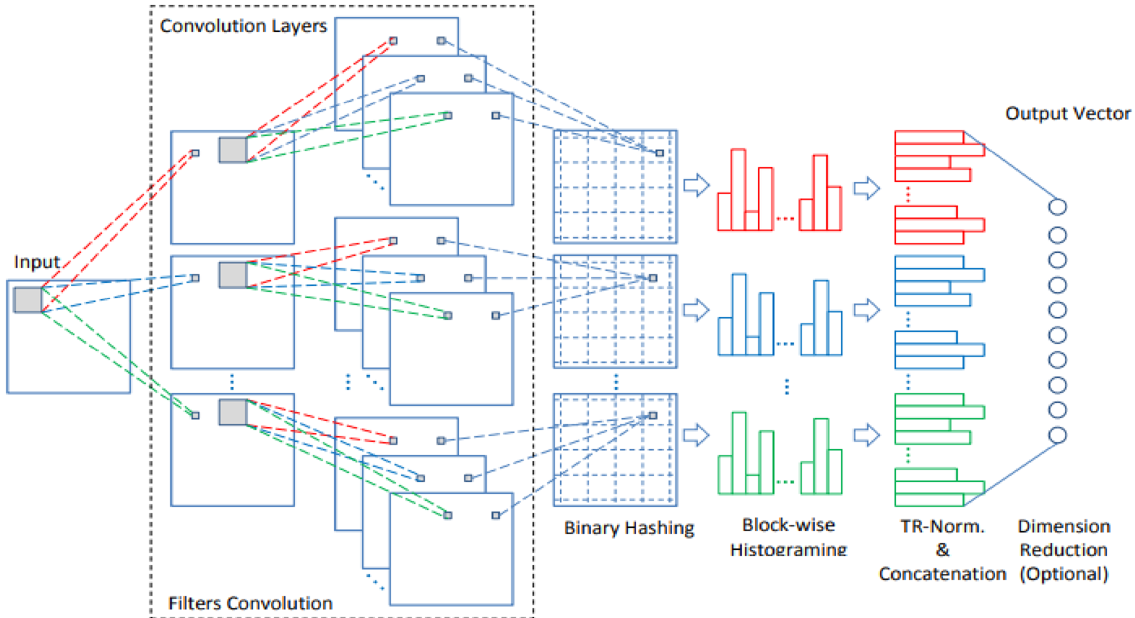

Figure 30 The block diagram of the proposed DCTNet.

1.1.1 Contribution 3: The Fusion of the CNN and the DCTNet:

We combined the CNN and the DCTNet by using a rule called "Mean" which works by doing a sum between the output of the CNN and the output of the DCTNet of the same class, and we used another rule called multiply which works by doing a multiplication between the two outputs from the same class, then we do a normalization to get a result in a vector of probabilities of each class.

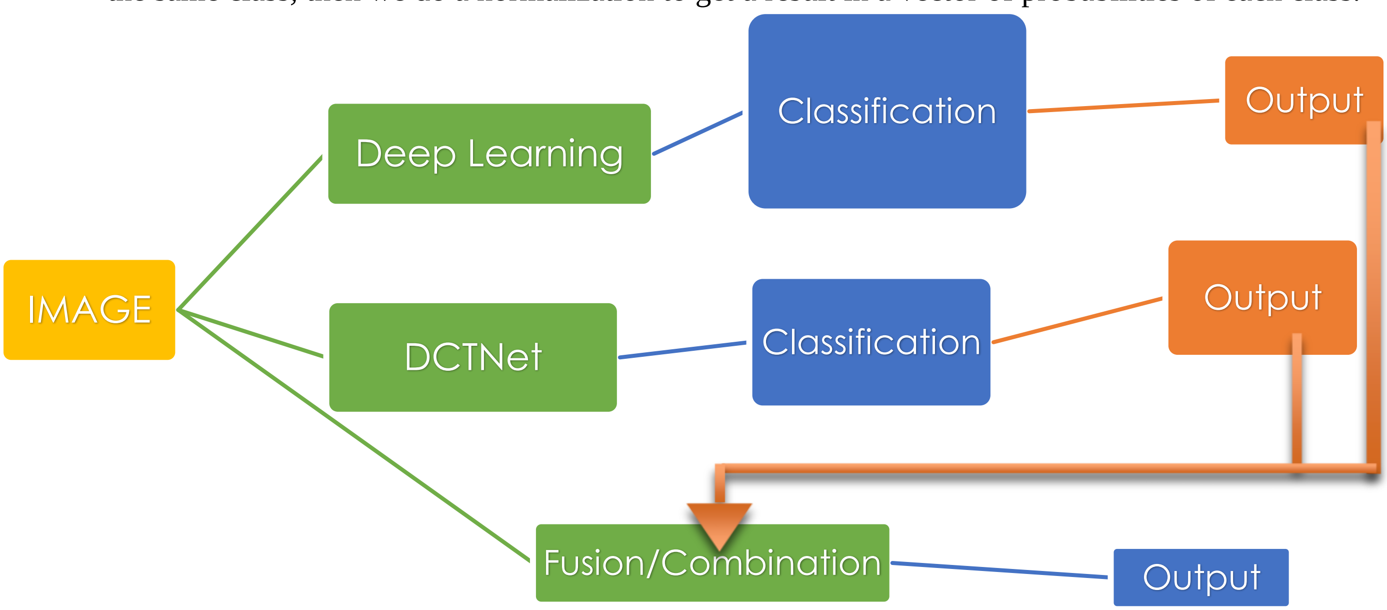

Figure 31 Our proposed Methods

1.2 Conclusion:

From the latest work with the CNN and DCTNet, we got many results some positive and some negative, which we will see in details in the next chapter.

We will devout the next chapter to the details of experimental results, with a description of the DATA used as well and the materials, followed by a discussion of the acquired results for each contribution.

4. Chapter 3: Results and discussions

4.1 Introduction:

Our contributions are done using the CNN and the DCTNet that explicitly described in Chapter 2.

While the first contribution concerns the use of CNN alone and the second one concerns the use of the DCTNet, the third targets a fusion of them both.

We devote this chapter to assess and discuss the achieved performance and efficiency of the proposed approaches.

For this purpose, we first start by a description of the dataset (section 1) as well as the materials (section 3) used, then in section 4 we delve, for each contribution assessment, into the acquired results followed by a discussion subsection.

4.2 Dataset:

In our subject ("the Breast cancer") we needed an accurate Dataset, that's why we choose out a Dataset from a website called Kaggle.

Our data reviews the medical images of breast cancer using ultrasound scan. Breast Ultrasound Dataset is categorized into three classes: normal, benign, and malignant images. Breast ultrasound images can produce great results in classification, detection, and segmentation of breast cancer when combined with machine learning.

Additionally, to the ultrasound images we access to a mask for each image where the tumor is well visible and pointed to, in some cases we didn't resort to using it.

The Dataset contains 1571 image including masks:

- Normal tumor : 265 Image, (133 ultrasound scan /132 Mask).
- Malignant : 417 Image, (210 ultrasound scan /207 Mask).
- Benign: 883 Image, (437 ultrasound scan /446 Mask).

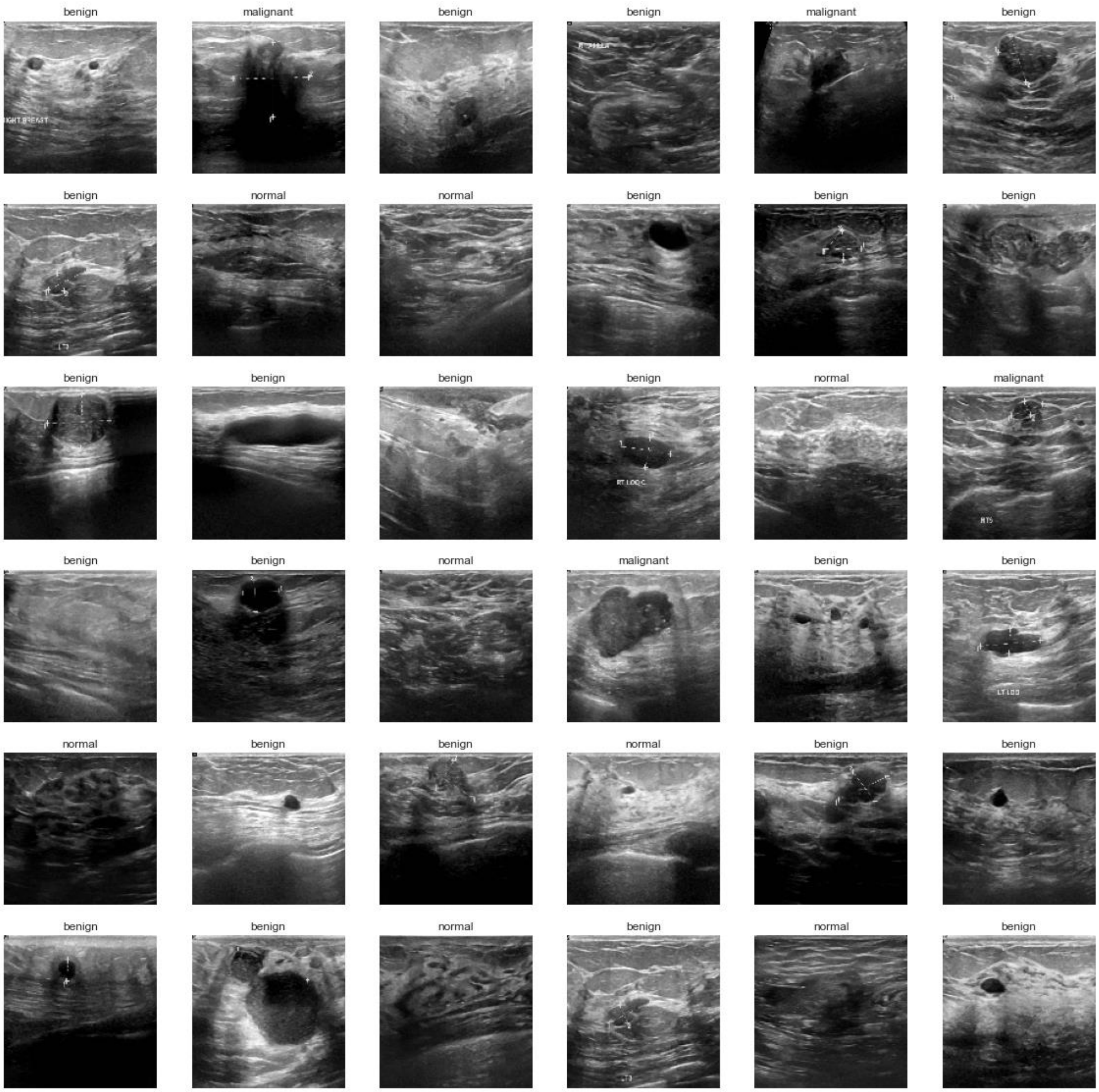

Figure 32 A brief look of the Dataset we used.

4.3 Materials:

- TensorFlow.
- Python.
- Keras.
- Scikit-learn.
- Jupyter.

4.4 Hardware:

All the executions were performed on two different laptops, it is based on regular Jupyter Notebook. The used hardwares have the following specifications:

1. ASUS X550CC:
    - Intel®core TM i7-3537V CPU @2.50GHz
    - 6G RAM
    - NVIDIA GeForce GT 720M
2. HP Notebook:
    - Intel®core TM i7-7500 CR CPU @ 2.70GHz
    - 8G RAM
    - AMD Radeon TM R7 M340

4.5 Results and Discussions:

In this section, the results of each contribution are discussed in details.

4.5.1 Contribution 1: The use of CNN:

As we said in the Chapter 2, after many changes we had many results

A. Processing:

We start by dividing the Dataset, to 90% for the training set and 10% for the validation set.

In some cases, we normalize each ultrasound image, by subtracting the mean and dividing by the standard deviation, to get a normalized ultrasound image with zero means and unit variance, and during some training procedures and to avoid over-fitting, we apply a data augmentation technique called ("Image DATA generator").

The Dataset was recognized as an RGB-images, we kept it this way.

B. Results:

By using multiple CNNs and different number of epochs we got the following results:

Table 1 Comparison between multiple CNNs used.

CNNs	Number of epochs	Accuracy train	Loss train	Accuracy test	Loss test
Model 1	200	0.9164	0.2058	0.7564	0.9095

Model 2	200	0.9181	0.1923	0.7692	0.9920
Xception+Flatten+Dense(softmax)	3	0.5587	0.9592	0.5769	2.5693
VGG16+Flatten+Dense(softmax)		0.5463	1.0007	0.5769	0.9736
VGG19+Flatten+Dense(softmax)		0.5623	0.9834	0.5769	0.9714
ResNet50+Flatten+Dense(softmax)		0.5356	2.6333	0.5769	1676992.1250
ResNet50V2+Flatten+Dense(softmax)		0.5801	1.0145	0.5769	748.0010
ResNet152+Flatten+Dense(softmax)		0.5552	3.0959	0.1667	116303981445120.0000
ResNet152V2+Flatten+Dense(softmax)		0.5498	1.3539	0.3718	41282720.0000
InceptionV3+ Flatten+Dense(softmax)		0.5872	0.9742	0.2692	34.6384
InceptionResNetV2+ Flatten+Dense(softmax)		0.5996	0.9733	0.5897	1.2660
MobileNet+ Flatten+Dense(softmax)		0.6655	1.9576	0.5897	4.0817
MobileNetV2+ Flatten+Dense(softmax)		0.6192	1.0665	0.5769	5.2603
DenseNet121+ Flatten+Dense(softmax)		0.5374	1.4232	0.5769	1889.4615
DenseNet169+ Flatten+Dense(softmax)		0.4982	1.4085	0.5769	9827.9756
DenseNet201+ Flatten+Dense(softmax)		0.5996	0.9802	0.5769	11463349.0000
EfficientNetB0+ Flatten+Dense(softmax)		0.6103	1.1054	0.2564	1.2650
EfficientNetV2B0+ Flatten+Dense(softmax)		0.5569	1.0308	0.5256	1.0081
EfficientNetB1+ Flatten+Dense(softmax)		0.5569	1.2417	0.2564	1.6141
EfficientNetV2B1+ Flatten+Dense(softmax)		0.5836	0.9425	0.2949	1.2325
EfficientNetB2+ Flatten+Dense(softmax)		0.6174	1.0306	0.2564	1.0893

EfficientNetV2B2+ Flatten+Dense(softmax)		0.5356	1.0875	0.2564	2.0664
EfficientNetB3+ Flatten+Dense(softmax)		0.6032	1.1164	0.2564	2.4866
EfficientNetV2B3+ Flatten+Dense(softmax)		0.5605	0.9873	0.5769	1.0064
EfficientNetB5+ Flatten+Dense(softmax)		0.6851	0.9397	0.1667	2.8940
Model 3	30	0.7794	0.5337	0.5769	2.2749
Mode 3 with freezing all the layers MobileNetV2	30	0.5623	0.9957	0.5769	1.0376
Model 4	3	0.4555	1.1167	0.5769	0.9855
Model 5	200	0.8416	0.3569	0.7179	0.6410
Mode 5 with freezing all the layers ResNet50	200	0.6246	0.8328	0.7051	0.7462
Model 5	1000	0.9733	0.0684	0.7949	0.9822
Model 6	100	0.9687	0.0670	0.8333	1.0926
Model 6 with the last 100 layers Train open	100	0.9915	0.0171	0.8462	0.8508

We made Model 1,2,3,4,5,6 on our own.

We noticed that the accuracy train in the model 4 (figure 51) and in the DenseNet169 is very low, but generally in the other models the accuracy is above 53%.

We noticed that the loss train is always between 0.0171 and 3.0959, the deference between all the models is minor. We got some massive loss test values, and the rest were acceptable.

C. Discussion:

We mostly depended on the loss and the accuracy of the test Data to determine the best model possible for our work, we tried to found the model with the lowest Loss and the highest accuracy, as we see in the Table 1, Model 6 with the last 100 layers train open is the best performing model we got.

4.5.2 Contribution 2: The use of DCTNet:

After generating 64 filter, each time we pick three filters randomly, and after following the DCTNet steps explained in the chapter 2, we had many results.

A. Processing:

We start by dividing the Dataset, to 90% for the training set and 10% for the validation set, in some occasion we used Dataset with the masks and sometimes we didn't.

We started by picking 3 random filters then we picked 7 filters using the zigzag scan.

We converted all the dataset into Grayscale-images.

We apply the DCTNet steps to it.

B. Results:

After changing multiple times, the filter and the classification algorithm.

Table 2 Best result for each algorithm

Algorithm	Filter used	Score test
KNN Cosine	**[[4, 7], [5, 3], [6, 2]]**	**56.41%**
KNN Euclidean	**[[4,4],[5,7],[6,2]]**	**<u>60.25%</u>**
Logistic Regression	**[[0,1],[1,0],[2,0],[2,2], [0,2],[0,3],[1,2]]**	**48.71%**
Support Vector classifier (SVC)	**random 3 filters**	**56.32%**
Naïve bayes Gaussian	**[[0, 4], [2, 1], [3, 7]]**	**50%**
Naïve bayes Multinomial	**[[0,0],[0,1],[1,0],[2,0], [2,2],[0,2],[0,3]]**	**50%**
Decision Tree	**[[2, 5], [3, 5], [7, 4]]**	**60.25%**

Random Forest Classifier	**[[0,0],[0,1],[1,0],[2,0],[2,2],[0,2],[0,3]]**	**57.69%**
Gradient Boosting	**[[0,0],[0,1],[1,0],[2,0],[2,2],[0,2],[0,3]]**	**48.71%**
Stochastic gradient descent Classifier	**[[0,1],[1,0],[2,0],[2,2],[0,2],[0,3],[1,2]]**	**48.71%**

We noticed that the accuracy changes depending on the changes of the number of filters used in the DCTNet and the Algorithm, and also the dataset with or without the mask got an impact too.

C. Discussion:

We depended on the score test to extract the best result, we found many similar results, the best result was when we used the decision tree and KNN Euclidean, the score was 60.25% for both.

Most of the results acquired in the first contribution were better than these results.

4.5.3 Contribution 3: The Fusion of them both:

After combining the last two with two rules “mean”, “Multiply” seen in chapter 2 we had many results.

A. Processing:

We took the same sample test from the CNN and the DCTNet to use it here. We used the best result acquired in the first contribution (Model 6 with 100 layers train open) and some of the used algorithms in the DCTNet.

B. Results:

The following table displays the results obtained:

Table 3 Results of the fusion

Accuracy CNN	Algorithm score test(Algorithm)	Accuracy DCTNet + CNN (Rule mean)	Accuracy DCTNet + CNN (Rule Multiply)
0.8462	3 filtres		
	0.5632911392405063 (SVC)	0.7974683544303798	
	0.5769230769230769 (KNN cosine)	0.782051282051282	
	0.6025641025641025 (KNN ecludiant)	0.782051282051282	
	0.48717948717948717(LoR)	0.9743589743589743	
	0.48717948717948717(NB G)	0.7564102564102564	
	0.5384615384615384 (SVC)	0.9743589743589743	
	7 filters		
	0.44936708860759494(KNN cosine)	0.8607594936708861	0.8227848101265823
	0.4177215189873418 (KNN ecludiant)	0.8734177215189873	0.8354430379746836
	0.5126582278481012 (SVM)	0.9113924050632911	0.9177215189873418
	0.43670886075949367(NB G)	0.6075949367088608	0.6075949367088608
	0.2974683544303797 (NB M)	0.33544303797468356	0.33544303797468356
	0.41139240506329117(SGDC)	0.759493670886076	0.759493670886076
	0.5189873417721519 (DTC)	0.7911392405063291	0.8037974683544303
	0.5759493670886076 (RFC)	0.7848101265822784	0.7911392405063291
	0.41139240506329117(GBC)	0.7974683544303798	0.810126582278481
	0.3987341772151899 (LoR)	0.8417721518987342	0.8417721518987342
	0.46153846153846156 (KNN cosine)	0.8846153846153846	0.8076923076923077
	0.4358974358974359 (KNN ecludiant)	0.8589743589743589	0.7948717948717948
	0.48717948717948717 (SVM)	0.9743589743589743	0.9743589743589743

	0.48717948717948717 (NB G)	0.6410256410256411	0.6410256410256411
	0.47435897435897434(NB M)	0.6794871794871795	0.6794871794871795
	0.48717948717948717 (SGDC)	0.7564102564102564	0.7564102564102564
	0.5 (DTC)	0.9743589743589743	0.9743589743589743
	0.5769230769230769 (RFC)	0.9743589743589743	0.9743589743589743
	0.48717948717948717 (GBC)	0.9743589743589743	0.9743589743589743
	0.48717948717948717(LoR)	0.9743589743589743	0.9743589743589743
	0.5 (KNN cosine)	0.8846153846153846	0.8076923076923077
	0.4230769230769231 (KNN ecludiant)	0.9102564102564102	0.8076923076923077
	0.5 (SVM)	0.9743589743589743	0.9743589743589743
	0.46153846153846156 (NB G)	0.6794871794871795	0.6794871794871795
	0.5 (NB M)	0.6923076923076923	0.6923076923076923
	0.48717948717948717 (SGDC)	0.7051282051282052	0.7051282051282052
	0.6025641025641025 (DTC)	0.9743589743589743	0.9743589743589743
	0.5769230769230769 (RFC)	0.9743589743589743	0.9743589743589743
	0.48717948717948717 (GBC)	0.9743589743589743	0.9743589743589743
	0.48717948717948717(LoR)	0.9871794871794872	0.9871794871794872

We can see that there are many good results.

C. Discussion:

After the fusion, we noticed that they have a huge impact on each other, when both of the them accuracies are good, the combination is the same or better, and also when one of the predicts in one of them is quite similar, the other one help it to do a better prediction.

for example, when we used the Decision tree in the second contribution, we got 60.25%, after the combination (with both rules (mean/multiply)) with the model 6 with 100 layers train open CNN, the accuracy increased to 97.43%.

another example, while the DCTNet result wasn't good enough 48.7%, after the combination with the CNN we found a Great result 98.71%, this one was because in general, the DCTNet wasn't good in the whole dataset but some samples were better classified than the CNN, that made the accuracy improve.

Overall, we can say that the fusion was a huge success.

D. Coefition matrix:

We can see that most of the predictions were correct, (44+20+13+=77) right ones, and 1 wrong one.

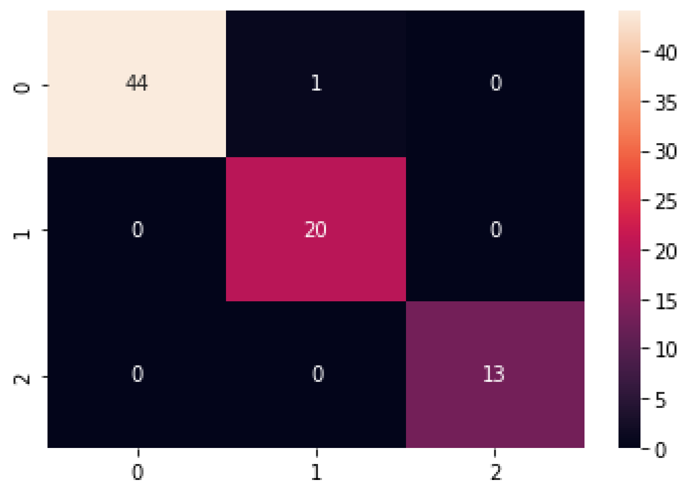

Figure 33 Coeftion Matrix of the best result acquired (Logistic regresioon)+CNN model.

4.6 Conclusion:

We started by describing the Dataset we used, then we dived into the materials, then we presented our first contribution, where we used Different CNN architectures in the process. The results showed that the use of CNN takes too much time and a lot of resources, which is not that helpful. On the other hand, in the second contribution, where we used the DCTNet, that showed that the use of DCTNet was lacking accuracy in comparison with the first contribution, but the process was faster and didn't require much. In the last contribution the results were good, but we hoped that the results obtained in the second contribution were better, because the results of the third contribution depend on the results of the first one which requires too much time and resources, we wanted less train time and less use of resources, we concluded the last contribution with an Evaluation matrix of the best result we obtained in our experiments. After reviewing all the results, we noticed that when the mask in the dataset wasn't included the results were much better.

5. Conclusion:

The use of artificial intelligence (AI) in diagnostic medical imaging is undergoing extensive evaluation. AI has shown impressive accuracy and sensitivity in the identification of imaging abnormalities and promises to enhance tissue-based detection and characterization [35].

Cancer detection and characterization presents a use for AI and ML in medical imaging. Inspired by the huge impact AI has on the medical field, we selected in this work, two main approaches, namely, a convolutional neural network (CNN), unsupervised DCTNet, to enhance early detection and classification of breast cancer.

We first started by exploring and implementing them from scratch in a dataset we found on Kaggle, to understand them well. The main goal of our contributions was to explore two different methods of classification, and to see if a combination between them would be beneficial. We used many CNN architectures in the first contribution, the train took too much time, and the process required heavy materials. The second was approach (DCTNet), was faster and much less consuming, but the results weren't that efficient. The last approach, which was a fusion of the first two was huge success, we didn't think that a combination would be rewarding.

We struggled with the lake of good materials, especially with the hardware, and we also had some small difficulties while understanding ML concepts.

We genially think that the use of ML in the medical field is still in its early stages, more and more beneficial upgrades are under develop, and we believe that the resources problem wouldn't be impactful in the future, because all the new machines are highly capable.

6. Références

1. ***Fundamentals of medical imaging.*** **Suetens : s.n., 2017.**

2. ***A review of machine learning and deep learning applications.*** **Shinde, P.P. 2018.**

3. Watson. ***Image compression using the discrete cosine transform.*** **1994. p.81.**

4. ***Understanding of a convolutional neural network.*** **lbawi, S., Mohammed. 2017.**

5. Peterson. ***K-nearest neighbor.*** **s.l. : Scholarpedia, 2009. p.1883.**

6. Bindushree, V., Sameen, R.J., Vasudevan, V., Shrihari, T.G., Devaraju, D. and Mathew, N.S., Artificial Intelligence: In Modern Dentistry.