Convolutional Neural Network for Medical Imaging Analysis: Abnormality detection in mammography

Barsellotti Luca, Emilio Paolini – A.A. 2020/21 – Computational Intelligence and Deep Learning

1 INDEX

2 INTRODUCTION

The objective of this project is to perform abnormality classification in mammography using Deep Learning techniques. The dataset provided is originated from the CBIS-DDSM: Curated Breast Imaging Subset of Digital Database for Screening Mammography. The original images are in DICOM format, but the provided dataset is composed by numpy arrays containing images and labels from training and test sets. From each image, two patches have been extracted:

1) Abnormality patch
2) Baseline patch

Then, to each patch, a class label has been assigned:

- 0: Baseline patch
- 1: Mass, benign
- 2: Mass, malignant
- 3: Calcification, benign
- 4: Calcification, malignant

So, the structure of the dataset is:

- Training set:
    - images tensor shape (5352,150,150)
    - labels tensor shape (5352,)
- Public Test set:
    - images tensor shape (672,150,150)
    - labels tensor shape (672,)

The classes distribution of images is:

- Train

	Benign	**Malignant**	**Total**
Train Masses	620	598	1218
Train Calcification	948	510	1458
Total	1568	1108	2676

- Test

	Benign	**Malignant**	**Total**
Global Test Masses	620	598	1218
Global Test Calcification	948	510	1458
Total	1568	1108	2676

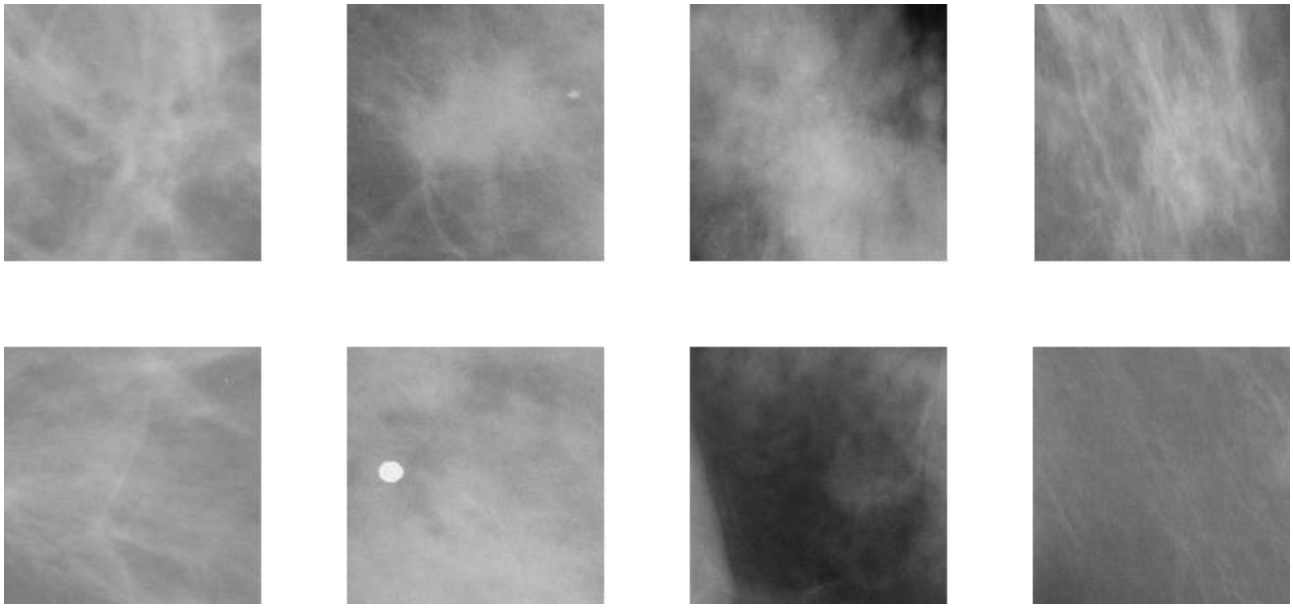

Figure 1: In the first row there are the abnormality patches and in the second row the baseline patches. From left to right: Mass/Benign, Mass/Malignant, Calcification/Benign and Calcification/Malignant.

There are two main tasks: to distinguish between “Mass” and “Calcification and to distinguish between “Benign” and “Malignant”. The provided solutions are all inspired by the literature.

2.1 REVIEW OF STATE-OF-THE-ART WORKS

There are several state-of-the-art works in the literature that try to solve the problem of the classification of mammograms. In this document, the most relevant, according to the task of classifying into "Benign"/"Malignant" or "Mass"/"Calcification", are exploited.

The first paper analysed is "An Ad Hoc Random Initialization Deep Neural Network Architecture for Discriminating Malignant Breast Cancer Lesions in Mammographic Images" (Duggento, et al., 2019) in which the authors proposed a CNN architecture to dicriminate between "benign" and "malignant" breast lesions. A combination of data augmentation and a simple model, that uses few pooling and convolutional layers, is used to perform the classification task. Despite the simplicity of the approach, the results can be considered promising, reaching an area under the receiver operatic characteristics curve of 0.785 and an accuracy equal to 71.19% on the test set.

The second paper analysed is "Optimizing and Visualizing Deep Learning for Benign/Malignant Classification in Breast Tumors" (Yi, et al., 2017) in which the authors challenge the "Benign"/"Malignant" classification task exploiting several techniques. First of all, they use as input both the standard mediolateral oblique (MLO) and craniocaudal (CC) views of the mammograms. This approach can be extended by considering the usage of the baseline patches. The images, before becoming inputs for the network, are pre-processed through a Difference of Gaussians approach, that is going to be exploited in 3.1.2 section. Moreover, the authors investigate two different deep CNN networks: a multi-modal network with two independent convolution branches and a network that has both MLO and CC images vote on the final prediction scores but share all weights in between, in a similar way to the siamese approach. The multi-modal is a network that creates two completely independent set of convolutional layers: one will take input from the CC images and the other will take input from the MLO images. At the end of the convolutional layers, we concatenate the flattened feature vectors and push them through a fully connected network.

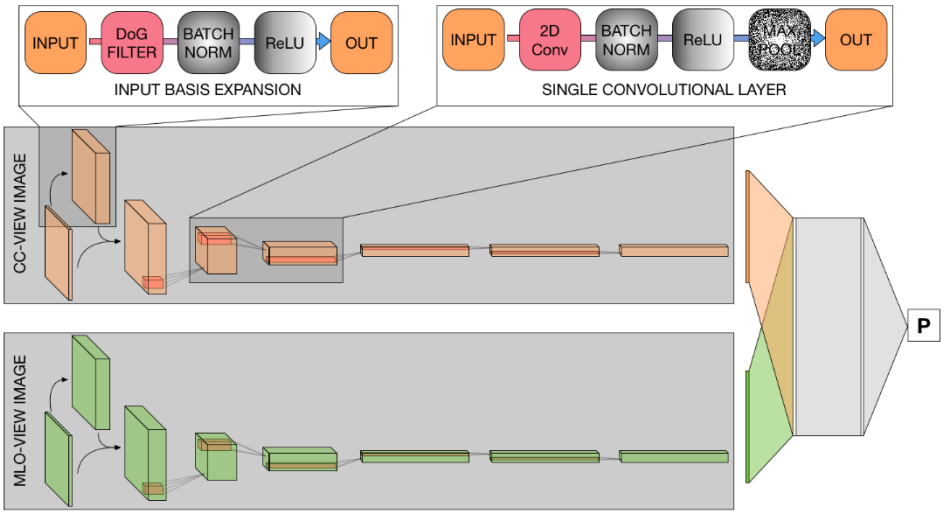

Figure 2: Multi-modal architecture proposed in "Optimizing and Visualizing Deep Learning for Benign/Malignant Classification in Breast Tumors" (Yi, et al., 2017).

The third paper analysed is "Five Classifications of Mammography Images Based on Deep Cooperation Convolutional Neural Network" (Tang, Cui, Yu, & Yang, 2019) in which the authors proposed propose a Deep Cooperation CNN (DCCNN) to classify mammography images of a data set into five categories including benign calcification, benign mass, malignant calcification, malignant mass and normal breast. The results, 91% accuracy and 0.98 AUC on the test

set, obtained by this type of network are superior to the ones obtained with VGG16, GoogLeNet and Inception V3. The architecture of the DCCNN is going to be analysed in section 3.1.3.

Another analysed paper is "Deep Learning to Improve Breast Cancer Detection on Screening Mammography" (Shen, et al., 2019) in which the authors test two pre-trained networks: VGG16 and ResNet50. The interesting aspect from this paper is how the authors perform the learning phase through three steps in which, at each step, only some layers are trained with different learning rates and the others are freezed. This approach is going to be analysed in section 4.1.2.

Finally, the last paper analysed is "Multi-tasking Siamese Networks for Breast Mass Detection Using Dual-View Mammogram Matching" (Yuton, et al., 2020) in which the authors combines craniocaudal (CC) and mediolateral-oblique (MLO) mammograms and pass them to a Siamese architecture with a deep-fusion approach, that concatenates the features extracted. Furthermore, the same structure but with difference of deep features has been exploited in section 5.1.1.

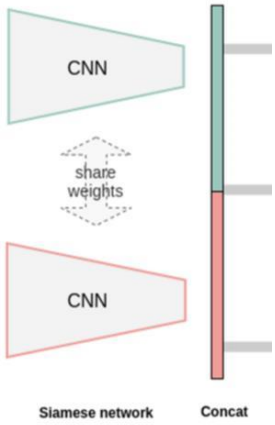

Figure 3: Siamese architecture with deep-fusion proposed in "Multi-tasking Siamese Networks for Breast Mass Detection Using Dual-View Mammogram Matching" (Yuton, et al., 2020).

3 SCRATCH MODEL

3.1 MASS VS CALCIFICATION

3.1.1 Experiment 1: Simple CNN

In the first experiment the simple Convolutional Neural Network proposed in "An Ad Hoc Random Initialization Deep Neural Network Architecture for Discriminating Malignant Breast Cancer Lesions in Mammographic Images " (Duggento, et al., 2019) is tested to implement the classification of the classes "Mass" and "Calcification". The data pre-processing is done only through a rescaling of 1/65535 factor to let all the images fall in the [0,1] range. This network is composed by:

1. Convolutional Layer with 64 filters 7x7
2. Dropout 25%
3. Max Pooling 4x4
4. Convolutional Layer with 64 filters 5x5

5. Dropout 25%
6. Max Pooling 3x3
7. Convolutional Layer with 64 filters 3x3
8. Dropout 25%
9. Max Pooling 2x2
10. Fully Connected Layer with 50 neurons and ReLU
11. Fully Connected Layer with 10 neurons and ReLU

The loss function is the binary cross entropy and the batch size is set to 32. Moreover, an Adam optimizer with learning rate 0.001 and a 20% validation split are used. The validation split allows to use an early stopping criteria based on the validation loss with a patience equal to 30.

The best results are obtained with:

- Epochs: 34
- Training accuracy: 86.52%
- Training loss: 0.3252
- Validation accuracy: 83.79%
- Validation loss: 0.4095
- Test accuracy: **79.46%**
- Test loss: 0.4917

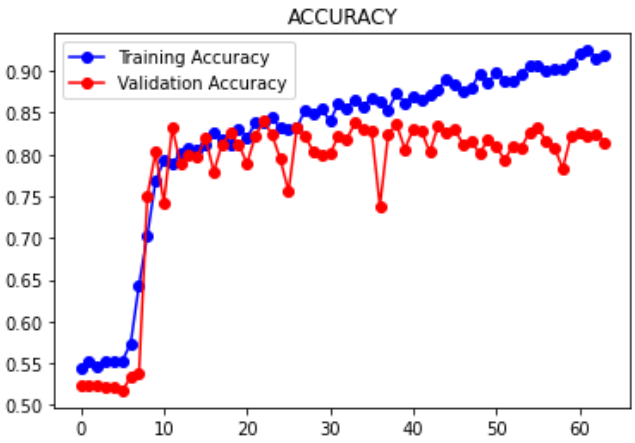

Figure 4: Comparison between the accuracy in function of the number of epochs on the training set and the validation set in experiment 1.

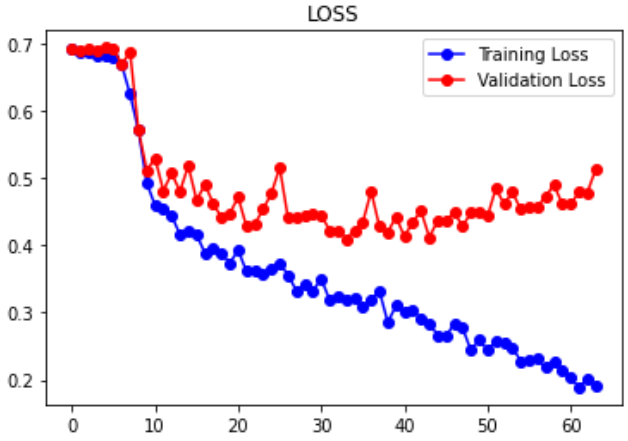

Figure 5: Comparison between the loss in function of the number of epochs on the training set and the validation set in experiment 1.

	Precision	**Recall**	**F1-score**
Mass	0.79	0.83	0.81
Calcification	0.80	0.75	0.77

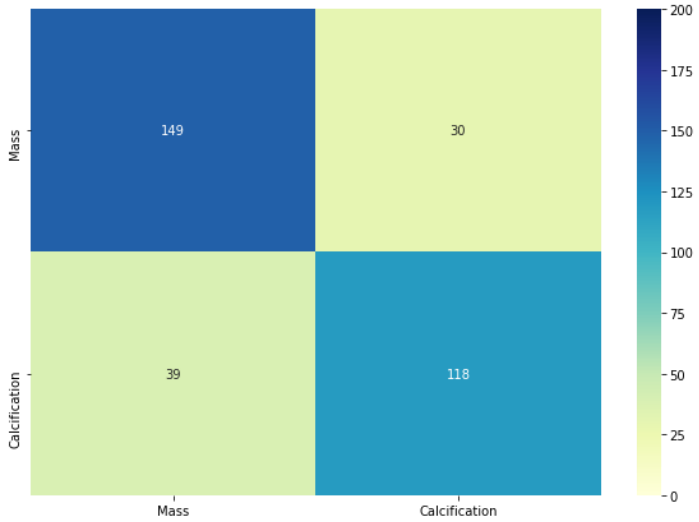

Figure 6: Confusion matrix computed on the test set using the experiment 1 model.

To improve the performance of this approach, two techniques are exploited:

1) **Data Augmentation**: the data augmentation has been done through random rotations in range [-40, 40], rescaling in range [0.8, 1.2], shear deformations in range [0.8, 1.2], width and height shifts in range [0.8, 1.2] and horizontal flipping.
2) **Regularization**: the L1 regularization with a 10^{-4} regularization factor in the Dense layers. This is used to limit the growth of the complexity of the model, in terms of weights.

These values for data augmentation and regularization are chosen after several tries in which the obtained results were affected by significant fluctuations. In particular, regularization used also during the Convolutional Layers caused this phenomenon.

The best results are obtained with:

- Epochs: 232
- Training accuracy: 84.26%
- Training loss: 0.3849
- Validation accuracy: 87.11%
- Validation loss: 0.3385
- Test accuracy: **85.12%**
- Test loss: 0.3728

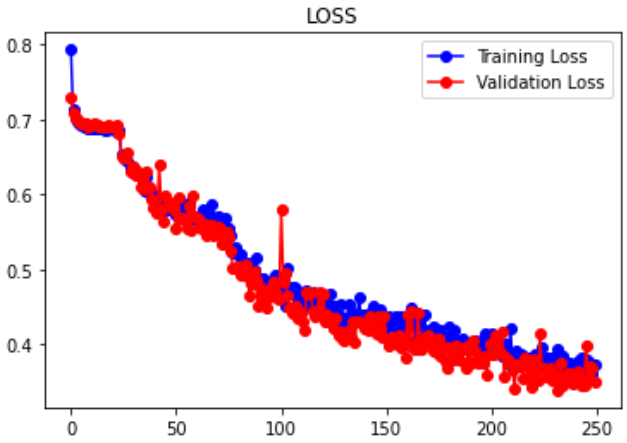

Figure 7: Comparison between the loss in function of the number of epochs on the training set and the validation set in experiment 1 (with data augmentation and regularization).

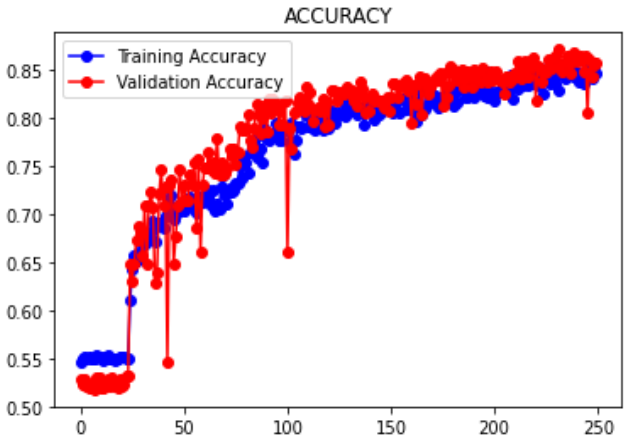

Figure 8: Comparison between the accuracy in function of the number of epochs on the training set and the validation set in experiment 1 (with data augmentation and regularization).

	Precision	**Recall**	**F1-score**
Mass	0.85	0.88	0.86
Calcification	0.85	0.82	0.84

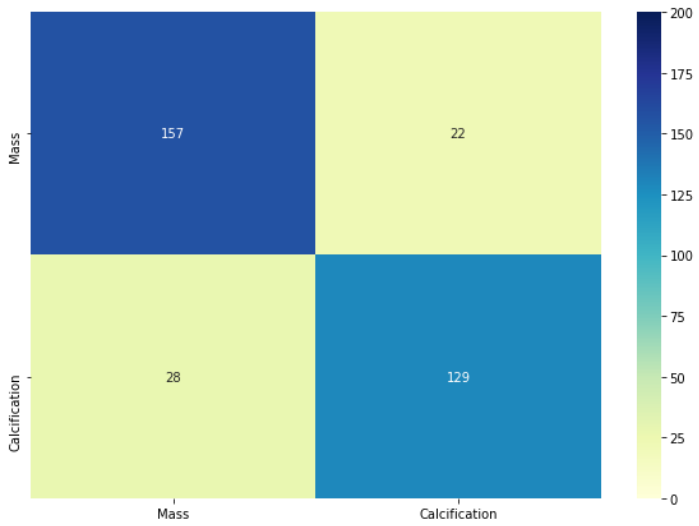

Figure 9: Confusion matrix computed on the test set using the experiment 1 model (with data augmentation and regularization).

3.1.2 Experiment 2: Exploiting Difference of Gaussians

The second experiment takes the model proposed in the first experiment but applies a further technique during the data pre-processing phase: Difference of Gaussians. As suggested in "Optimizing and Visualizing Deep Learning for Benign/Malignant Classification in Breast Tumors" (Yi, et al., 2017), Difference of Gaussians are used in Computer Vision approaches to find cornerns, blobs and regions. Applying a Gaussian filter on an image corresponds to blurring the image: the strenght of the blurring is given by the value of the standard deviation of the Gaussian. Getting the difference between two blurred images, got through two different Gaussian filters, allows to keep only the points in the image where the contrast is strongly changing and, consequently, understanding which points are parts of keypoints.

A single Difference of Gaussians is defined as:

$$DoG(\sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} - \frac{1}{\pi\sigma^2} e^{-\frac{x^2+y^2}{\sigma^2}}$$

A set of 8 Difference of Gaussians ($\sigma = \{\sqrt{2}, 2, 2\sqrt{2}, 4, 4\sqrt{2}, 8, 8\sqrt{2}, 16\}$ is computed on each image and given as input to one single CNN. Then, all the outputs from the different CNN are concatenated and elaborated through two Dense layers. At the end, the classification is performed using a binary cross entropy.

Each model assigned to a Difference of Gaussians (or to the original image) has different weights from the others. The best results are obtained with:

- Epochs: 192
- Training accuracy: 86.81%
- Training loss: 0.3817
- Validation accuracy: 90.73%
- Validation loss: 0.3257
- Test accuracy: **85.42%**
- Test loss: 0.3882

The test accuracy is a bit higher in this experiment (85.42% vs 85.12%) but the validation loss is a bit lower in experiment 1 (0.3728 vs 0.3882). It is possible to conclude that exploiting the Difference of Gaussians doesn't provide a significant change in performance.

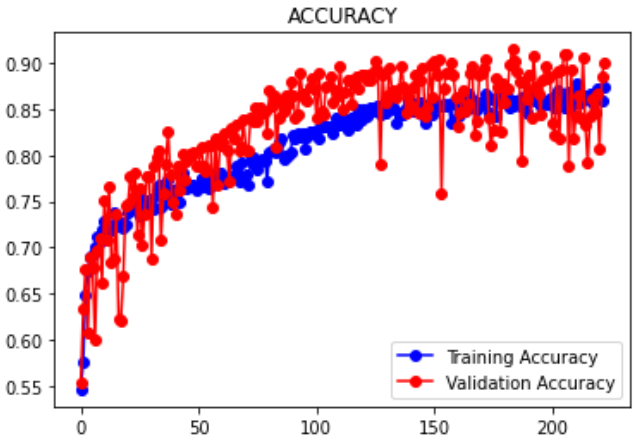

Figure 10: Comparison between the accuracy in function of the number of epochs on the training set and the validation set in experiment 2.

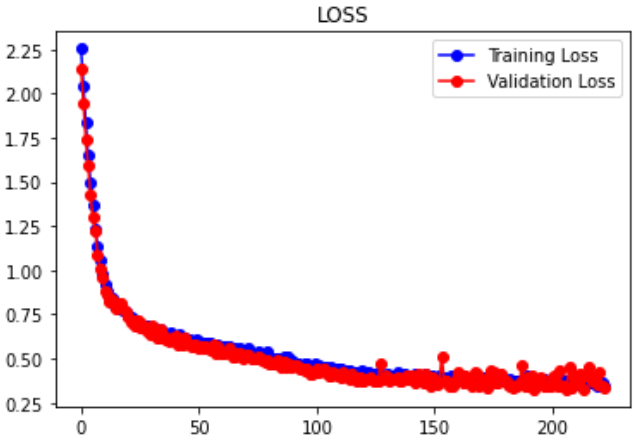

Figure 11: Comparison between the loss in function of the number of epochs on the training set and the validation set in experiment 2.

	Precision	**Recall**	**F1-score**
Mass	0.87	0.85	0.86
Calcification	0.83	0.86	0.85

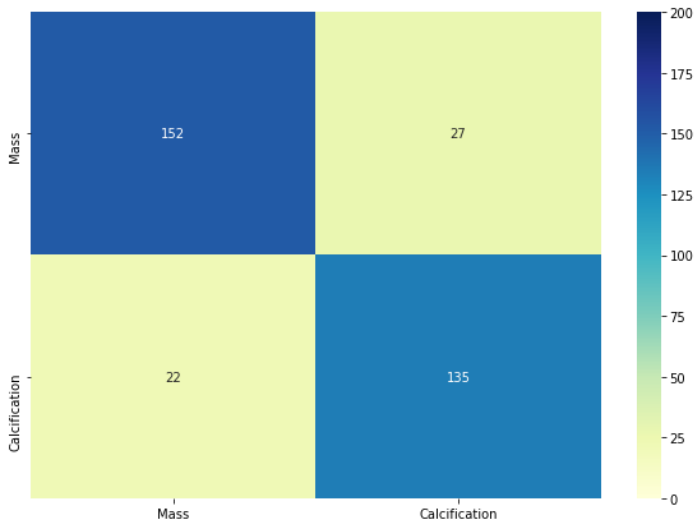

Figure 12: Confusion matrix computed on the test set using the experiment 2 model.

3.1.3 Experiment 3: DCCNN v1

The third experiment exploits the Deep Cooperation Convolutional Neural Network, as mentioned in "Five Classifications of Mammography Images Based on Deep Cooperation Convolutional Neural Network" (Tang, Cui, Yu, & Yang, 2019). The network proposed in the paper is composed by two CNN that work in parallel with the same structure but with different parameters, generating a sort of ensembling because the two DCCNN learn differently how to classify the inputs. The two outputs coming from the two networks are concatenated and given as input to a Fully Connected Layer.

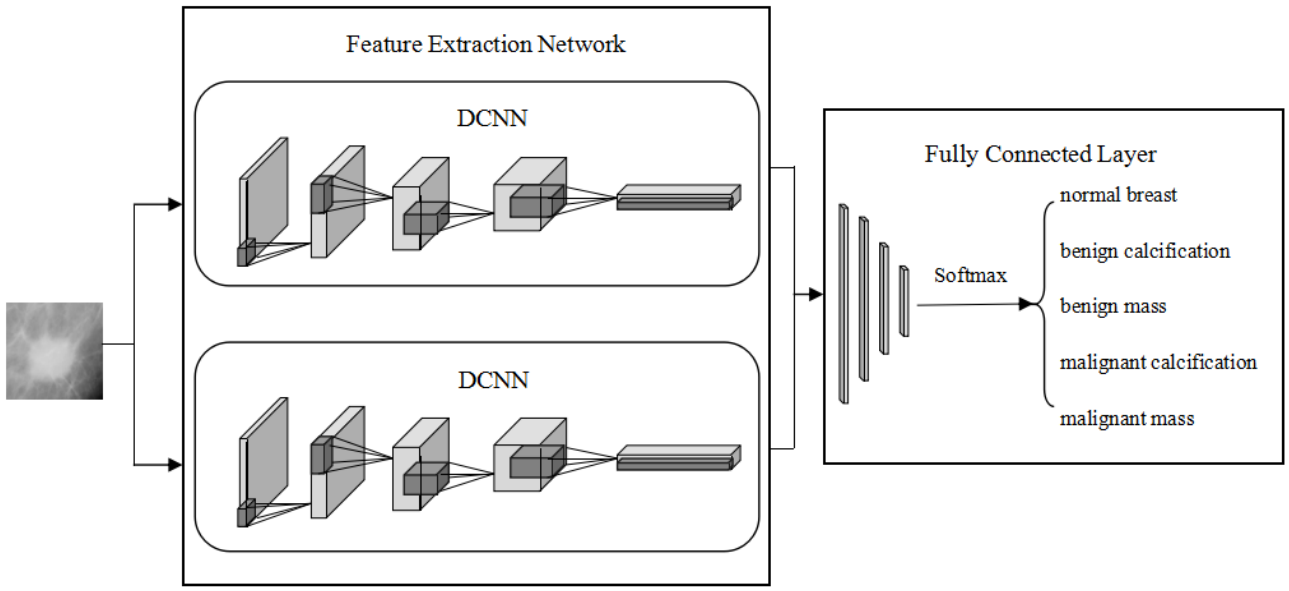

Figure 13: Representation of the architecture proposed by Tang et al..

Each DCNN is composed by:

1. Convolution 3x3x32
2. Max Pooling 2x2
3. Convolution 3x3x32
4. Convolution 1x1x64
5. Max Pooling 2x2

6. Convolution 1x1x80
7. Convolution 3x3x192
8. Max Pooling 2x2
9. Inception module 6a with 224 output depth
10. Inception module 6a with 224 output depth
11. Inception module 6b with 512 output depth
12. Inception module 7a with 512 output depth
13. Inception module 7a with 512 output depth
14. Inception module 7b with 896 output depth
15. Average Pooling 8x8 on layer 14
16. Average Pooling 34x34 on layer 6
17. Average Pooling 17x17 on layer 9
18. Average Pooling 17x17 on layer 12
19. Concatenate layer 14, 15, 16 and 17 (output: 1x1x1824)

Then, the Fully Connected Layer is composed by:

1. Dense 1024
2. Dense 512
3. Dense 32

The classification is done on the last Dense Layer using the binary cross entropy. Each layer uses the ReLU activation function. Moreover, a L2 regularization is used and the data augmentation is implemented through horizontal and vertical flipping, 90 degrees random rotations (with a random range of 10 degrees) and rescaling in range [0.7, 1.3]. The classes are weighted giving a higher weight to the minority class ("Mass"). The inception module 6a and 7a maintain the same height and width of the input volume, instead the 6b and 7b modules halve them.

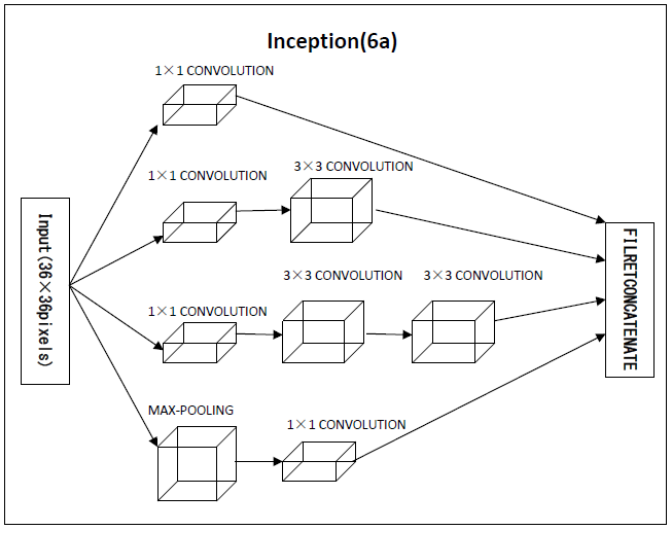

Figure 14: Representation of the architecture of an Inception 6a module.

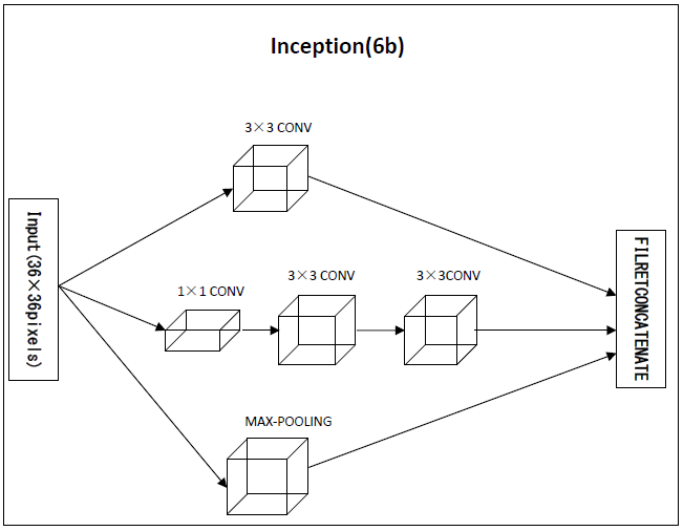

Figure 15: Representation of the architecture of an Inception 6b module.

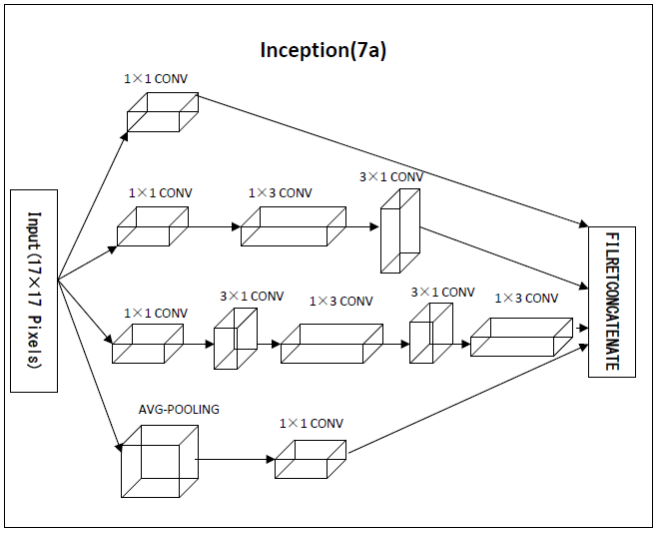

Figure 16: Representation of the architecture of an Inception 7a module.

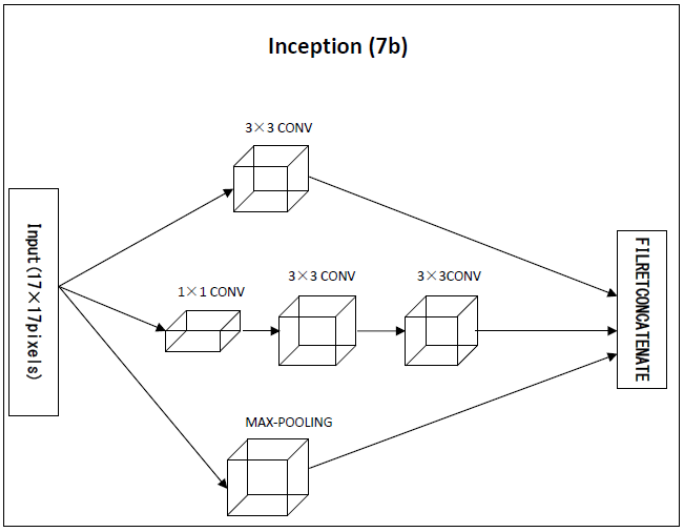

Figure 17: Representation of the architecture of an Inception 7b module.

The input image used in the paper is 299x299x1, so there are two possible solution:

1) **DCCNN v1**: Removing the first Max Pooling Layer and changing the second Convolutional Layer to a 2x2x32 kernel to keep the same dimensions among the network.
2) **DCCNN v2**: Keeping the same architecture but replacing the Average Pooling Layers with Global Average Pooling Layers. This is done to adapt the average pooling to the dimension of the input volume and to give always 1x1 as activation map size in output.

The best results for DCCNN v1 are obtained with:

- Epochs: 168
- Training accuracy: 91.72%
- Training loss: 0.2085
- Validation accuracy: 92.58%
- Validation loss: 0.2006
- Test accuracy: **88.69%**
- Test loss: 0.3119

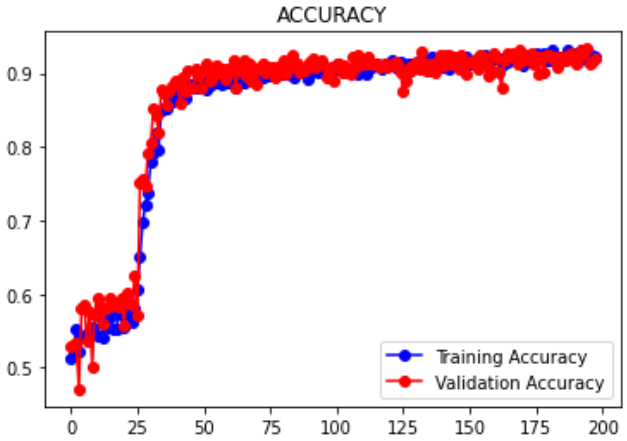

Figure 18: Comparison between the accuracy in function of the number of epochs on the training set and the validation set in experiment 3 (DCCNN v1).

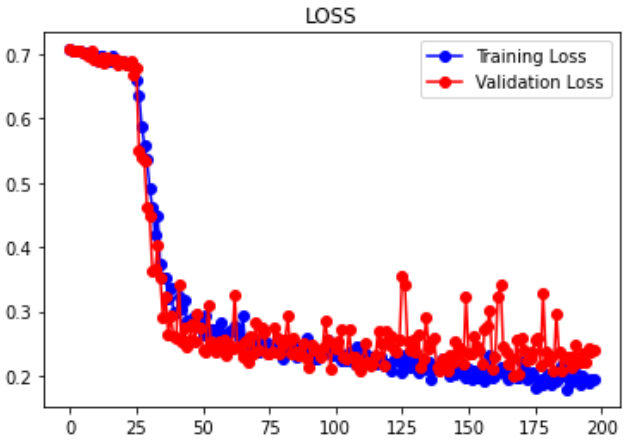

Figure 19: Comparison between the loss in function of the number of epochs on the training set and the validation set in experiment 3 (DCCNN v1).

	Precision	**Recall**	**F1-score**
Mass	0.88	0.92	0.90
Calcification	0.90	0.85	0.88

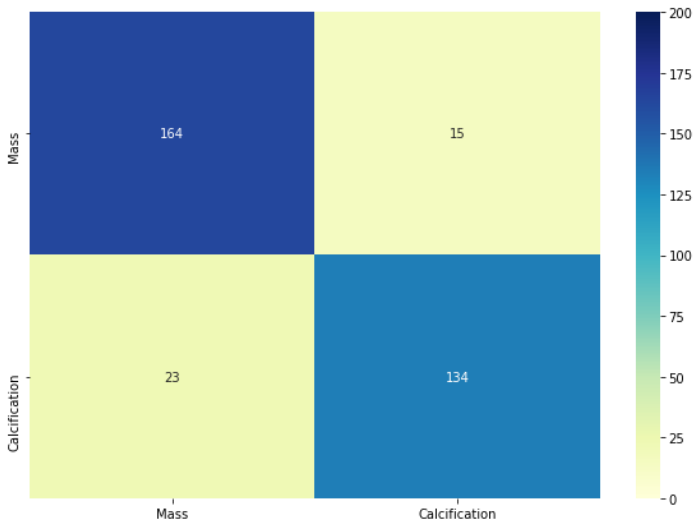

Figure 20: Confusion matrix computed on the test set using the experiment 3 model (DCCNN v1).

The best results for DCCNN v2 are obtained with:

- Epochs: 109
- Training accuracy: 92.77%
- Training loss: 0.1952
- Validation accuracy: 89.45%
- Validation loss: 0.2506
- Test accuracy: **87.20%**
- Test loss: 0.2839

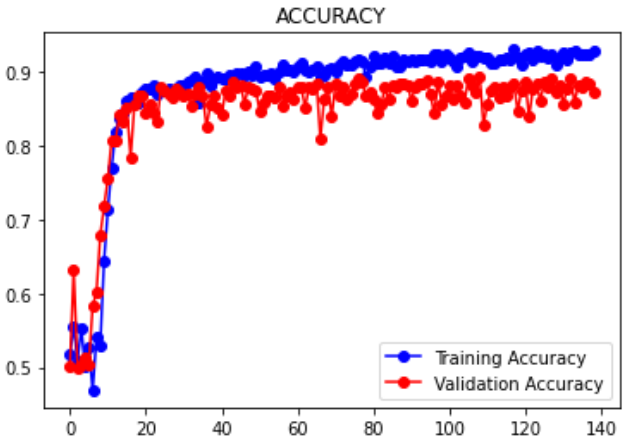

Figure 21: Comparison between the accuracy in function of the number of epochs on the training set and the validation set in experiment 3 (DCCNN v2).

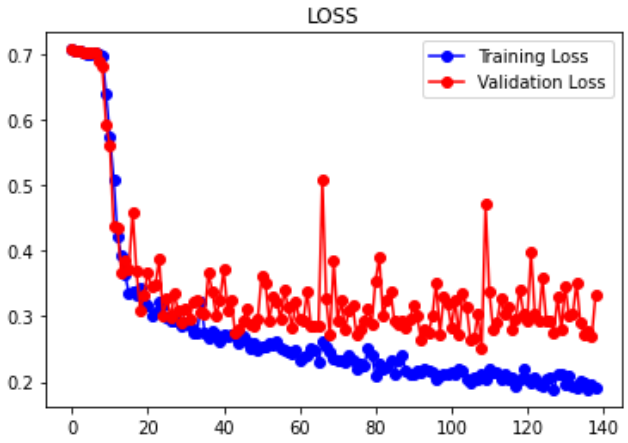

Figure 22: Comparison between the loss in function of the number of epochs on the training set and the validation set in experiment 3 (DCCNN v2).

	Precision	**Recall**	**F1-score**
Mass	0.84	0.93	0.89
Calcification	0.91	0.80	0.85

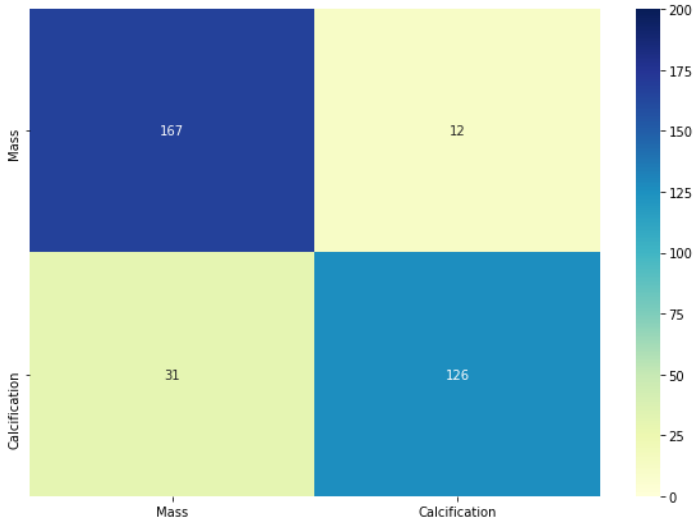

Figure 23: Confusion matrix computed on the test set using the experiment 3 model (DCCNN v2).

3.1.4 Results

To choose the best model from the previous experiments, the ROC curves and F1-scores are considered since classification of both classes is equally important.

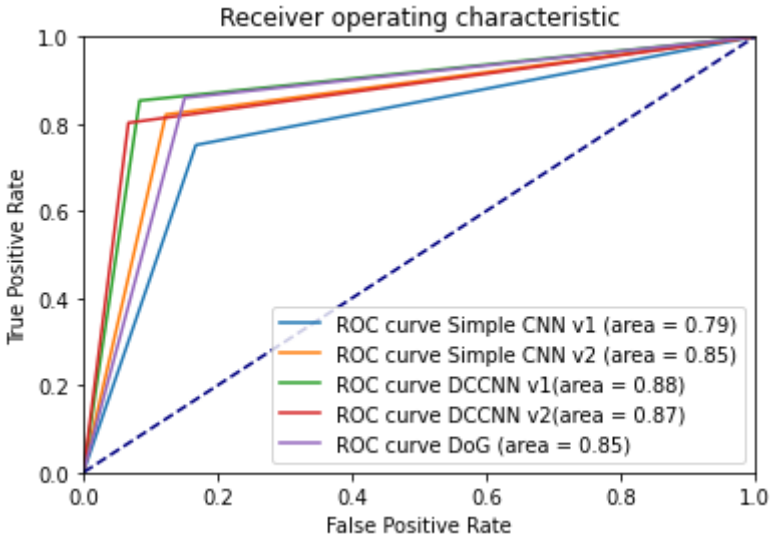

Figure 24: Comparison between the ROC curves computed on the four models based on "Mass"/"Calcification" classification.

	Accuracy Test	**F1-Score**	**Epochs**
Simple CNN v1	0.7946	0.79	34
Simple CNN v2	0.8512	0.85	232
DCCNN v1	**0.8869**	**0.89**	168
DCCNN v2	0.8720	0.87	109
DoG	0.8542	0.85	192

From these results, it is possible to conclude that the DCCNN v1 is the best model according to both F1-score and AUC. For this reason, later on this model has been considered the best one from scratch experiments.

The difference between the two versions of the DCCNN models is not significant, but the first version performs slightly better so it is taken as the reference model from these experiments.

3.2 BENIGN VS MALIGNANT

3.2.1 Experiment 4: Simple CNN

In this experiment the model proposed in section 3.1.1 (Simple CNN) is used with the same data augmentation and regularization, changing the classification task from “Mass”/”Calcification” to “Benign”/”Malignant”.

The best results are obtained with:

- Epochs: 113
- Training accuracy: 65.56%
- Training loss: 0.6057
- Validation accuracy: 67.19%
- Validation loss: 0.5949
- Test accuracy: **67.56%**
- Test loss: 0.6116

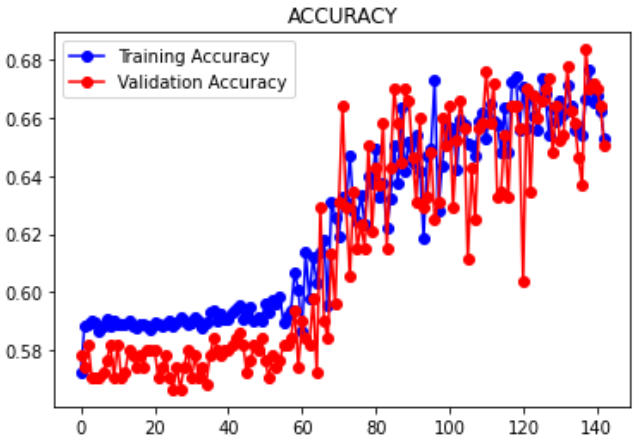

Figure 25: Comparison between the accuracy in function of the number of epochs on the training set and the validation set in experiment 5.

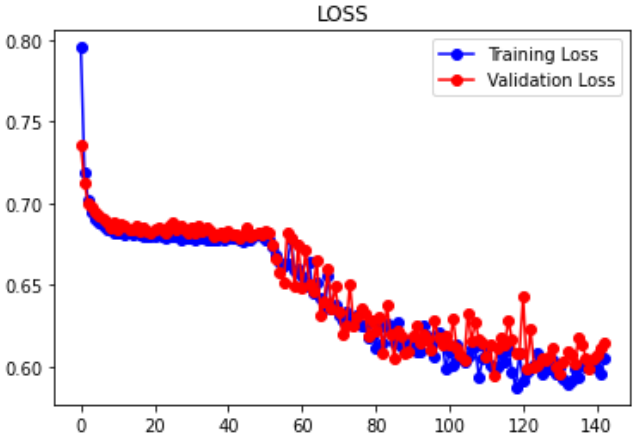

Figure 26: Comparison between the loss in function of the number of epochs on the training set and the validation set in experiment 5.

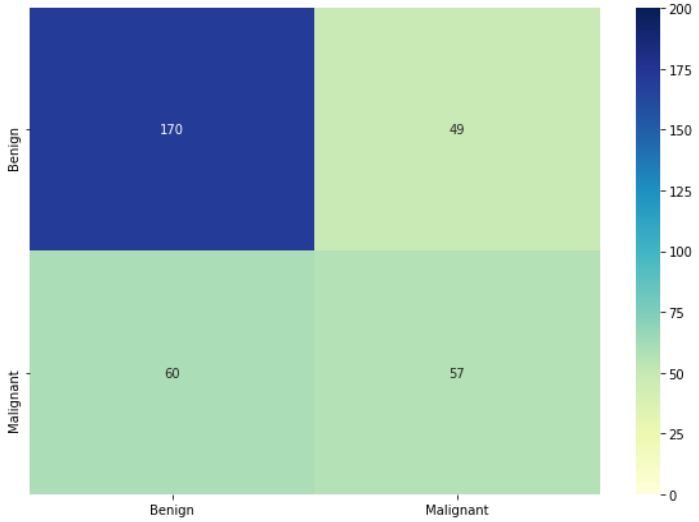

Figure 27: Confusion matrix computed on the test set using the experiment 5 model.

	Precision	**Recall**	**F1-score**
Benign	0.74	0.78	0.76
Malignant	0.54	0.49	0.51

3.2.2 Experiment 5: Exploiting Difference of Gaussians

In this experiment the input got through the 8 Difference of Gaussians applied to the image, as in section 3.1.2, is used to check if it might help during the recognition of "Benign" or "Malignant" classes.

The best results are obtained with:

- Epochs: 48
- Training accuracy: 66.77%
- Training loss: 0.6066
- Validation accuracy: 66.22%
- Validation loss: 0.6000
- Test accuracy: **67.86%**
- Test loss: 0.6161

As it happens in "Mass"/"Calcification" recognition, the difference in performance between the approach with the Difference of Gaussians and the approach without the Difference of Gaussians is not significantly different.

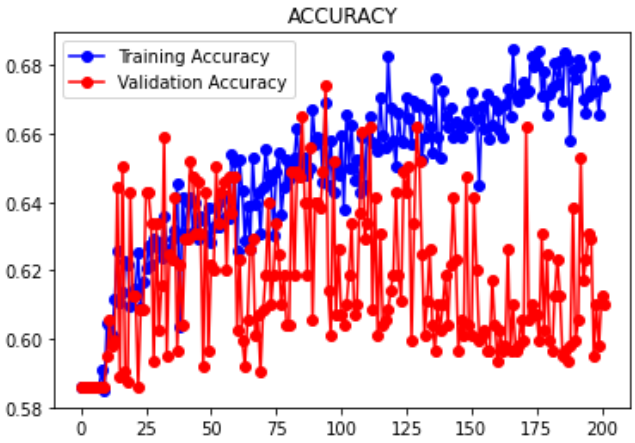

Figure 28: Comparison between the accuracy in function of the number of epochs on the training set and the validation set in experiment 6.

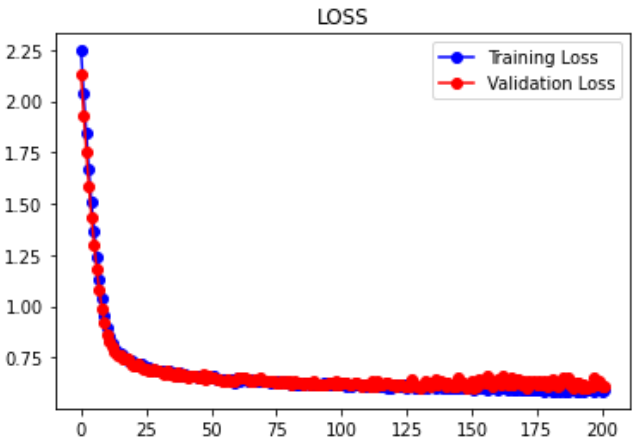

Figure 29: Comparison between the loss in function of the number of epochs on the training set and the validation set in experiment 6.

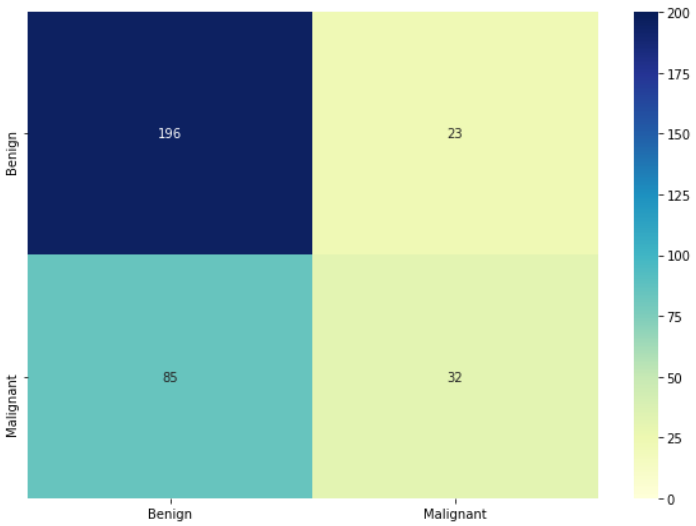

Figure 30: Confusion matrix computed on the test set using the experiment 6 model.

	Precision	**Recall**	**F1-score**
Benign	0.70	0.89	0.78
Malignant	0.58	0.27	0.37

3.2.3 Experiment 6: DCCNN v1

In this experiment the model proposed in section 3.1.3 is used, changing the classification task from "Mass"/"Calcification" to "Benign"/"Malignant".

The best results are obtained with:

- Epochs: 48
- Training accuracy: 65.80%
- Training loss: 0.6014
- Validation accuracy: 70.12%
- Validation loss: 0.5917
- Test accuracy: **64.58%**
- Test loss: 0.6270

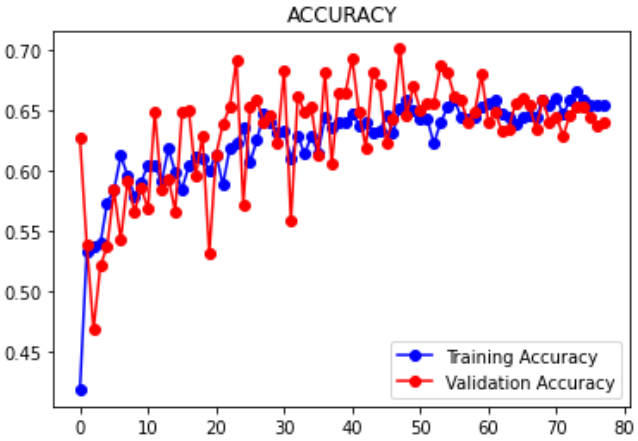

Figure 31: Comparison between the accuracy in function of the number of epochs on the training set and the validation set in experiment 6.

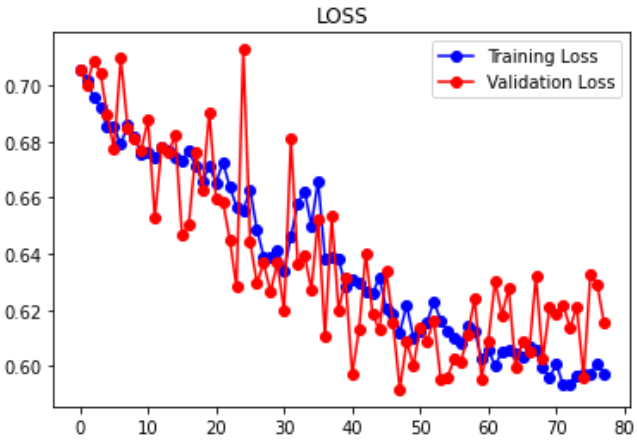

Figure 32: Comparison between the loss in function of the number of epochs on the training set and the validation set in experiment 6.

	Precision	**Recall**	**F1-score**
Benign	0.77	0.66	0.71
Malignant	0.49	0.62	0.55

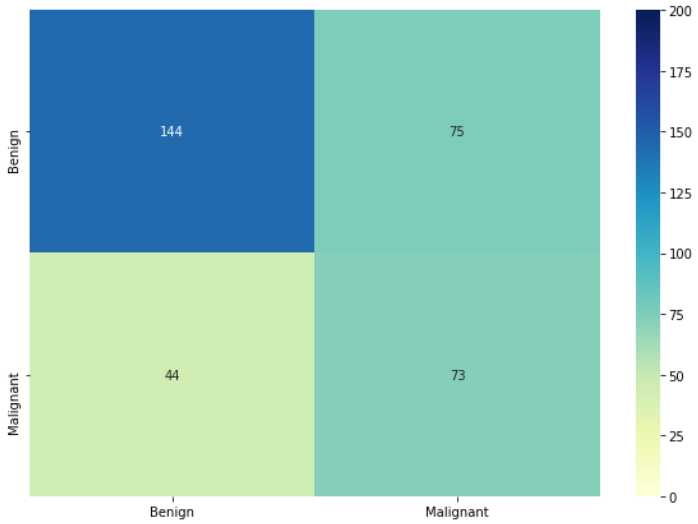

Figure 33: Confusion matrix computed on the test set using the experiment 6 model.

3.2.4 Results

Finally, the results from the three models are evaluated. The first comparison has been done using the ROC curve. Results can be seen in the figure below.

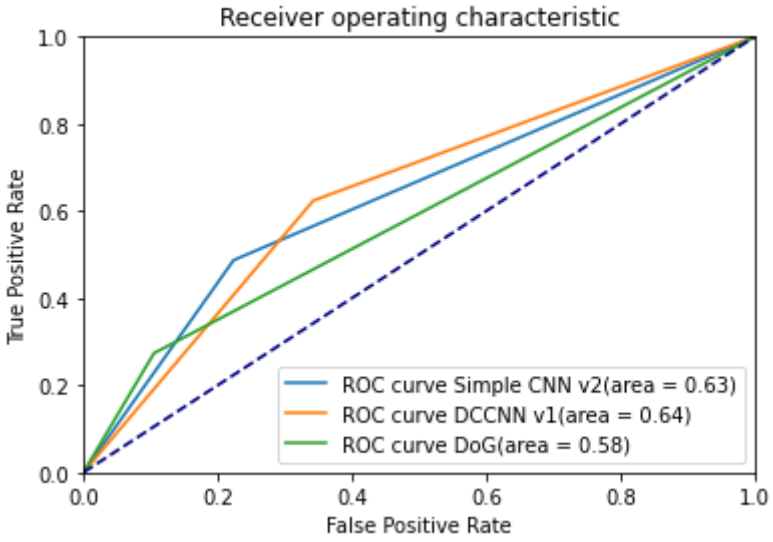

Figure 34: Comparison between the ROC curves computed on the two models based on "Benign"/"Malignant" classification.

Simple CNN v2 and DCCNN v1 outperform DoG, in terms of AUC.

Then, another important reasoning is the one about recognizing malignant (both mass and calcification) abnormality: we should consider a better model the one that can recognize the highest number of malignant abnormalities. This is due from the fact that misclassifying a benignant abnormality into a malignant one should be consider less risky than doing the opposite. Indeed, a benign tumour does not have the ability to invade neighbouring tissue while a malignant one can spread to other parts of the body through metastasize. For this reason, the detection of malignant tumour should be as soon as possible, in order to treat it when it is very small and curable.

According to the previous reasoning, F2 score is considered to identify the best model for "Benign"/"Malignant" classification. Moreover, recall on the malignant class is considered, since it answers the following question: "What proportion of actual positives was identified correctly?". So, having a higher recall implies identifying more instances of the malignant class.

The results obtained are reported in the following table:

	Recall Malignant	**F2-score**	**Epochs**
Simple CNN v2	0.49	0.49	113
DCCNN v1	**0.62**	**0.59**	48
DoG	0.27	0.30	48

Both in terms of recall and F2-score, the DCCNN v1 outperforms the other two models.

However, the obtained results (from the DCCNN v1) cannot be considered good, since it can only recognize 62% of malignant instances.

4 PRETRAINED MODEL

In this section, state-of-the-art pretrained models are used to perform both classification tasks.

4.1 MASS VS CALCIFICATION

4.1.1 Experiment 7: VGG16 not trainable

In "Five Classifications of Mammography Images Based on Deep Cooperation Convolutional Neural Network" (Tang, Cui, Yu, & Yang, 2019), three pre-trained models are tested for classification: VGG16, InceptionV3 and GoogLeNet. In this experiment, the VGG16 model pre-trained on the ImageNet dataset is used. After the model, two Dense layers with, respectively, dimension 512 and 256 are added. Only these new layers are trained and the VGG16 is only used as feature extractor. The classification is performed using a binary cross entropy loss. The data augmentation and the class weighting are performed as in section 3.1.3 (DCCNN v1).

The best results are obtained with:

- Epochs: 178
- Training accuracy: 86.53%
- Training loss: 0.2971
- Validation accuracy: 89.65%
- Validation loss: 0.2559
- Test accuracy: **84.23%**
- Test loss: 0.3293

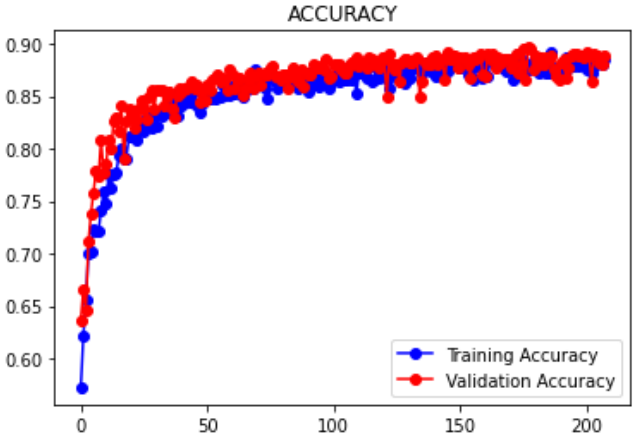

Figure 35: Comparison between the accuracy in function of the number of epochs on the training set and the validation set in experiment 7.

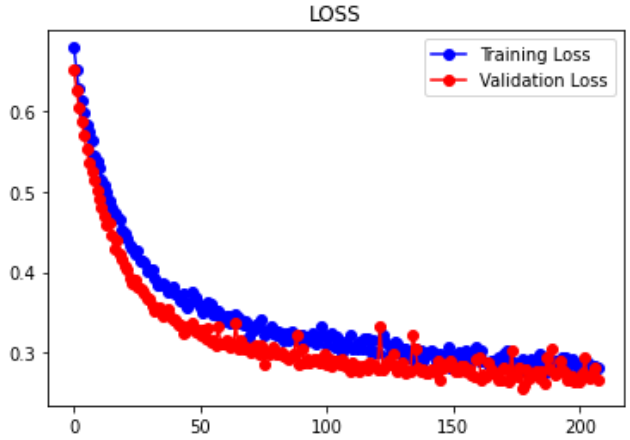

Figure 36: Comparison between the loss in function of the number of epochs on the training set and the validation set in experiment 7.

	Precision	Recall	F1-score
Mass	0.85	0.86	0.85
Calcification	0.84	0.82	0.83

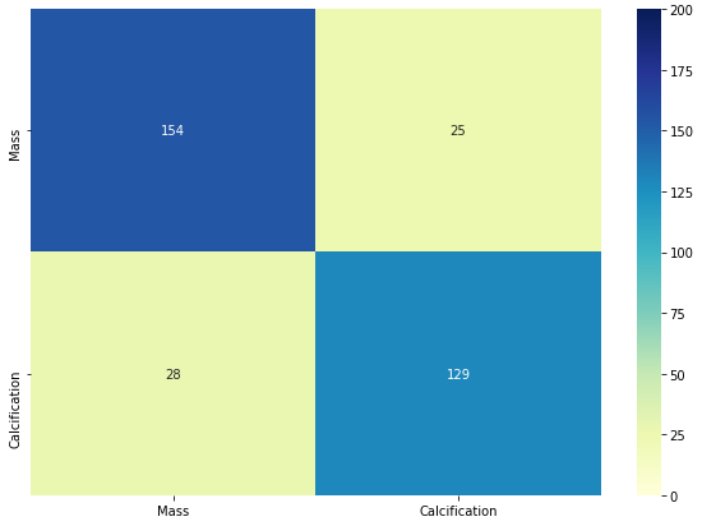

Figure 37: Confusion matrix computed on the test set using the experiment 7 model.

4.1.2 Experiment 8: VGG16 trainable

In the eighth experiment, the VGG16 with same architecture, data augmentation and class weighting of the previous experiment is used. The difference is that in this experiment all the layers are trained through a three-step learning phase as reported in the paper "Deep Learning to Improve Breast Cancer Detection on Screening Mammography" (Shen, et al., 2019):

1) Set learning rate to 10^{-3} and train the last Dense layers for 3 epochs.
2) Set learning rate to 10^{-4} and unfreeze the last 11 layers of the VGG16 and train for 10 epochs.
3) Set learning rate to 10^{-5} and unfreeze all the layers and train for 87 epochs (but select the best result basing on the validation loss).

The best results are obtained with:

- Epochs: 83
- Training accuracy: 96.44%
- Training loss: 0.1009
- Validation accuracy: 94.92%
- Validation loss: 0.1375
- Test accuracy: **89.58%**
- Test loss: 0.2564

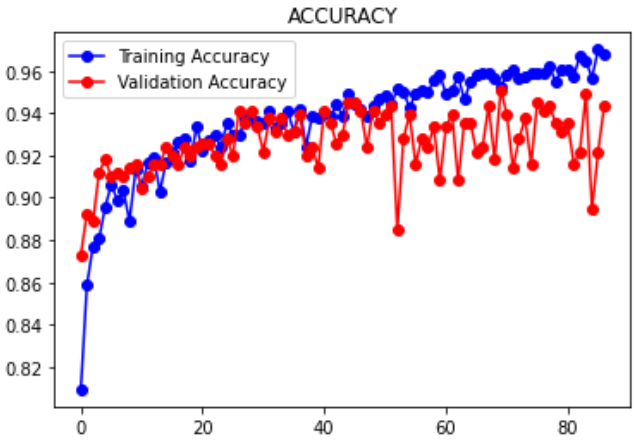

Figure 38: Comparison between the accuracy in function of the number of epochs on the training set and the validation set in experiment 8.

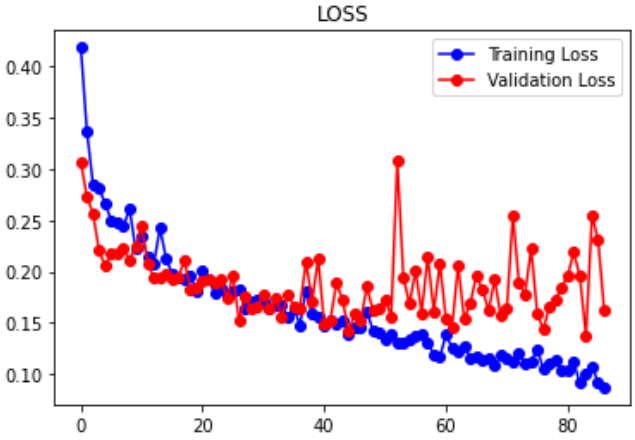

Figure 39: Comparison between the loss in function of the number of epochs on the training set and the validation set in experiment 8.

	Precision	**Recall**	**F1-score**
Mass	0.90	0.91	0.90
Calcification	0.89	0.89	0.89

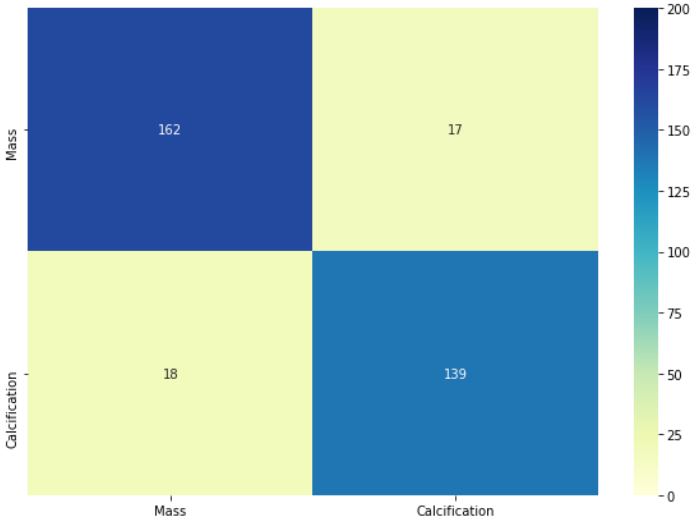

Figure 40: Confusion matrix computed on the test set using the experiment 8 model.

4.1.3 Experiment 9: InceptionV3 not trainable

In this experiment, still according to "Five Classifications of Mammography Images Based on Deep Cooperation Convolutional Neural Network" (Tang, Cui, Yu, & Yang, 2019), the use of InceptionV3 pre-trained on ImageNet is exploited. After the model, two Dense layers with, respectively, dimension 512 and 256 are added. Only these new layers are trained and the InceptionV3 is only used as feature extractor. The classification is performed using a binary cross entropy loss. The data augmentation and the class weighting are performed as in section 3.1.3 (DCCNN v1).

The best results are obtained with:

- Epochs: 30
- Training accuracy: 89.80%
- Training loss: 0.2404
- Validation accuracy: 88.67%
- Validation loss: 0.2755
- Test accuracy: **86.90%**
- Test loss: 0.3199

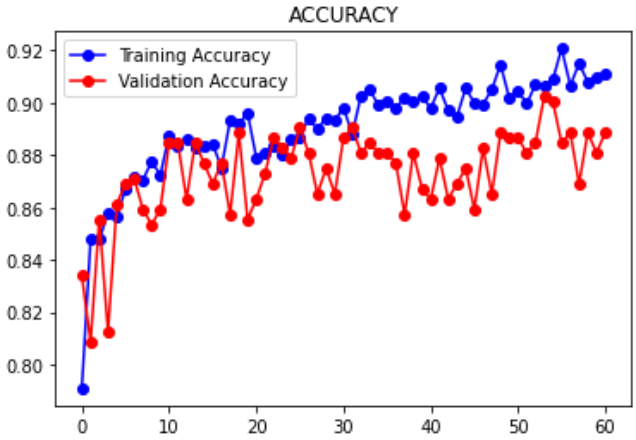

Figure 41: Comparison between the accuracy in function of the number of epochs on the training set and the validation set in experiment 9.

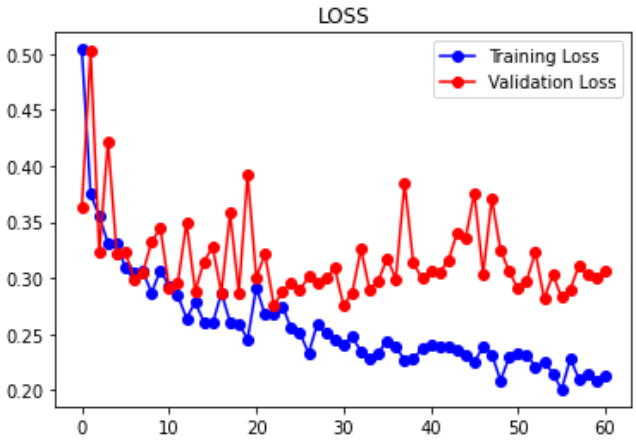

Figure 42: Comparison between the loss in function of the number of epochs on the training set and the validation set in experiment 9.

	Precision	**Recall**	**F1-score**
Mass	0.90	0.85	0.87

Calcification	0.84	0.89	0.86

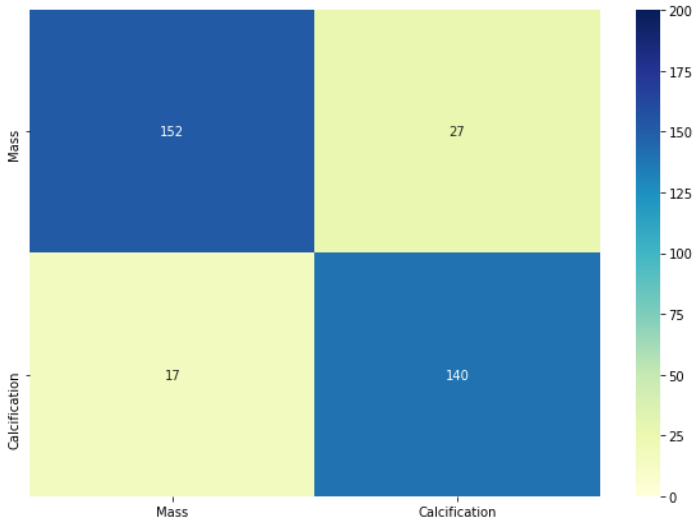

Figure 43: Confusion matrix computed on the test set using the experiment 9 model.

4.1.4 Experiment 10: InceptionV3 trainable

In this experiment, InceptionV3 with same architecture, data augmentation and class weighting of the previous experiment is used. The difference is that in this experiment all the layers are trained using a learning rate equal to 10^{-5}.

The best results are obtained with:

- Epochs: 49
- Training accuracy: 95.49%
- Training loss: 0.1243
- Validation accuracy: 90.23%
- Validation loss: 0.2745
- Test accuracy: **89.29%**
- Test loss: 0.2941

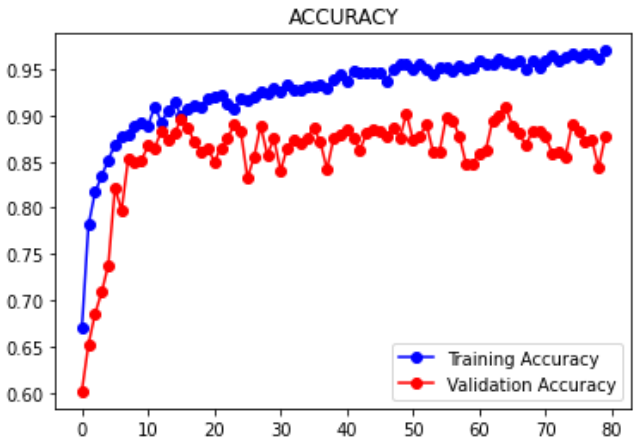

Figure 44: Comparison between the accuracy in function of the number of epochs on the training set and the validation set in experiment 10.

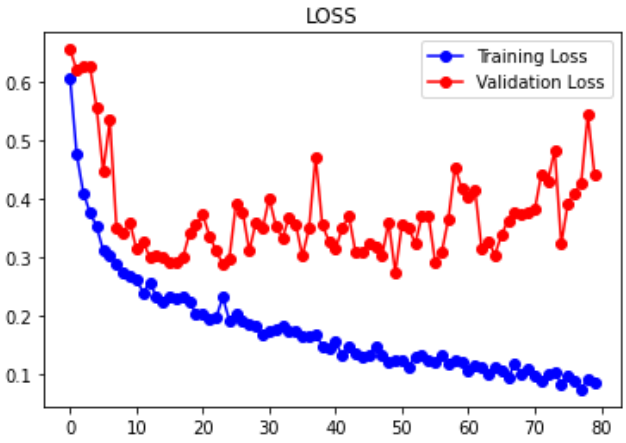

Figure 45: Comparison between the loss in function of the number of epochs on the training set and the validation set in experiment 10.

	Precision	**Recall**	**F1-score**
Mass	0.91	0.89	0.90
Calcification	0.88	0.90	0.89

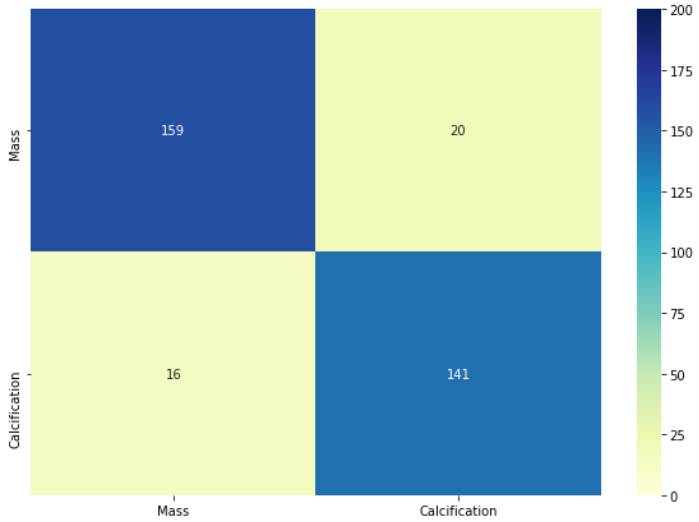

Figure 46: Confusion matrix computed on the test set using the experiment 10 model.

4.1.5 Results

As for the scratch models, ROC curves and F1-scores are considered as metrics to choose the best model.

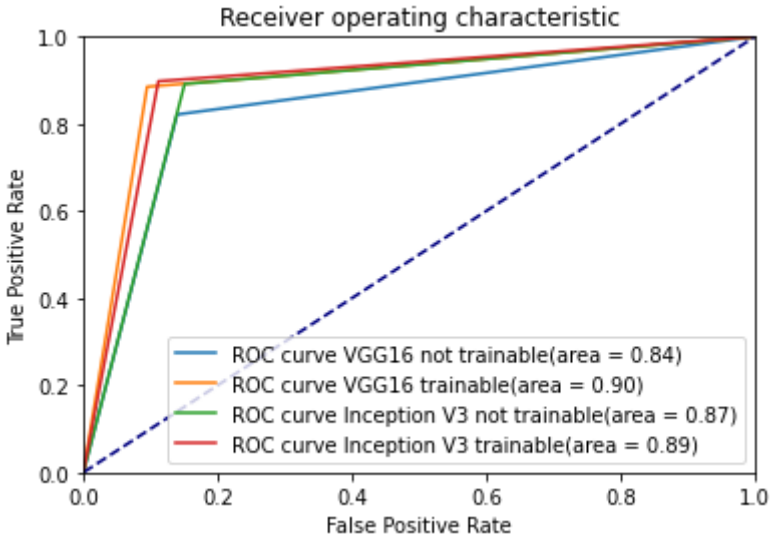

Figure 47: Comparison between the ROC curves computed on the four models based on "Mass"/"Classification" classification.

	Accuracy Test	**F1-Score**	**Epochs**
VGG16 not trainable	0.8423	0.84	178
VGG16 trainable	**0.8958**	**0.90**	83
Inception V3 not trainable	0.8690	0.87	30
Inception V3 trainable	0.8929	0.89	49

The VGG16 model, trained using the three-learning phase, outperforms the other tested models.

Anyway, the results obtained with the trainable Inception V3 are not significantly different. Both experiments can be considered as valid models for the classification task.

Also the ResNet50 pre-trained network has been tested, using the three step learning approach proposed in "Deep Learning to Improve Breast Cancer Detection on Screening Mammography" (Shen, et al., 2019), but it resulted in a failure because the network was not learning properly from the training set and for that reason the experiment is not reported.

4.2 BENIGN VS MALIGNANT

Best two models of previous section are used to perform classification on "Benign"/"Malignant".

4.2.1 Experiment 11: VGG16 trainable

In this experiment the model proposed in section 4.1.2 is used, changing the classification task from "Mass"/"Calcification" to "Benign"/"Malignant".

The best results are obtained with:

- Epochs: 25
- Training accuracy: 75.25%
- Training loss: 0.4792
- Validation accuracy: 76.17%
- Validation loss: 0.4879
- Test accuracy: **72.32%**

- Test loss: 0.5677

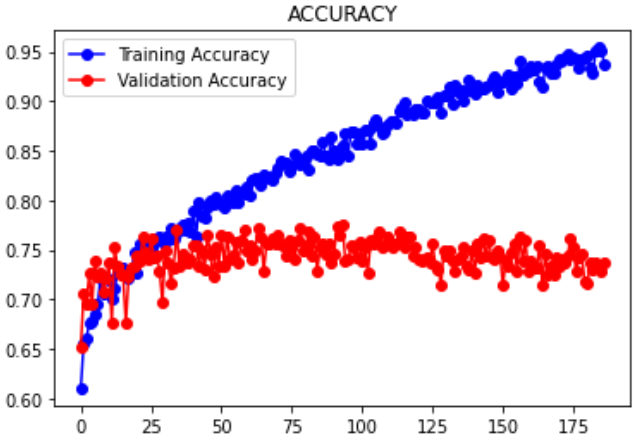

Figure 48: Comparison between the accuracy in function of the number of epochs on the training set and the validation set in experiment 11.

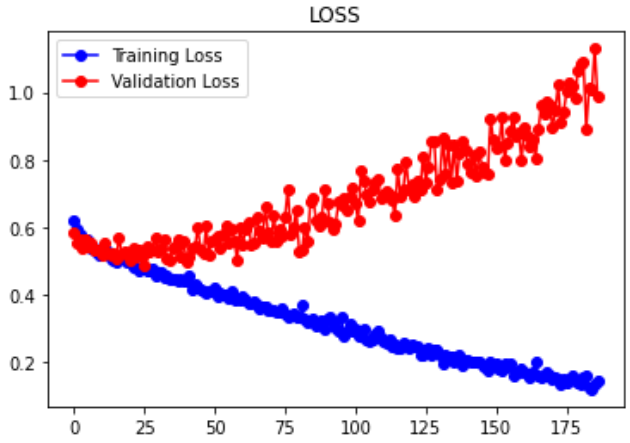

Figure 49: Comparison between the loss in function of the number of epochs on the training set and the validation set in experiment 11.

	Precision	**Recall**	**F1-score**
Benign	0.81	0.74	0.78
Malignant	0.59	0.68	0.63

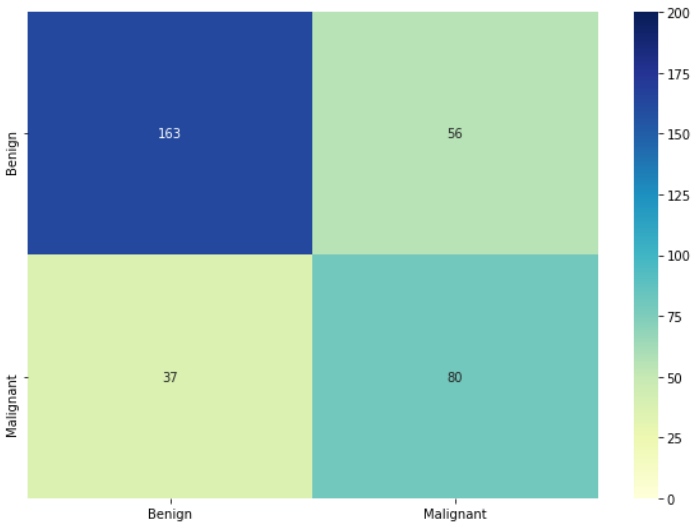

Figure 50: Confusion matrix computed on the test set using the experiment 11 model.

4.2.2 Experiment 12: InceptionV3 trainable

In this experiment the model proposed in section 4.1.4 is used, changing the classification task from "Mass"/"Calcification" to "Benign"/"Malignant".

The best results are obtained with:

- Epochs: 13
- Training accuracy: 71.36%
- Training loss: 0.5333
- Validation accuracy: 70.89%
- Validation loss: 0.5759
- Test accuracy: **66.07%**
- Test loss: 0.5984

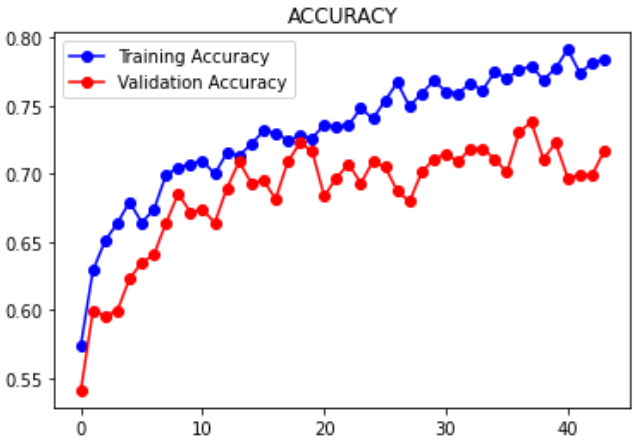

Figure 51: Comparison between the accuracy in function of the number of epochs on the training set and the validation set in experiment 12.

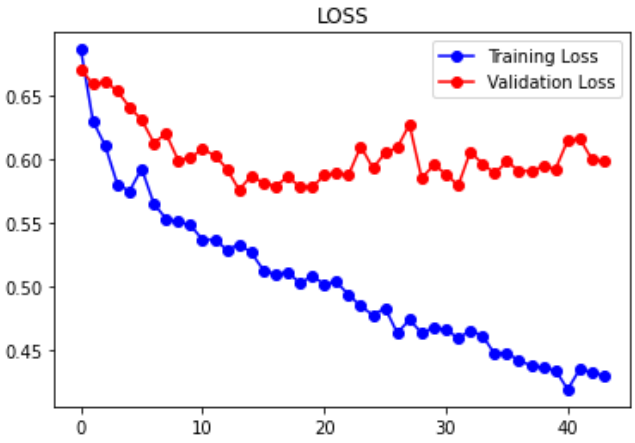

Figure 52: Comparison between the loss in function of the number of epochs on the training set and the validation set in experiment 12.

	Precision	**Recall**	**F1-score**
Benign	0.80	0.64	0.71
Malignant	0.51	0.70	0.59

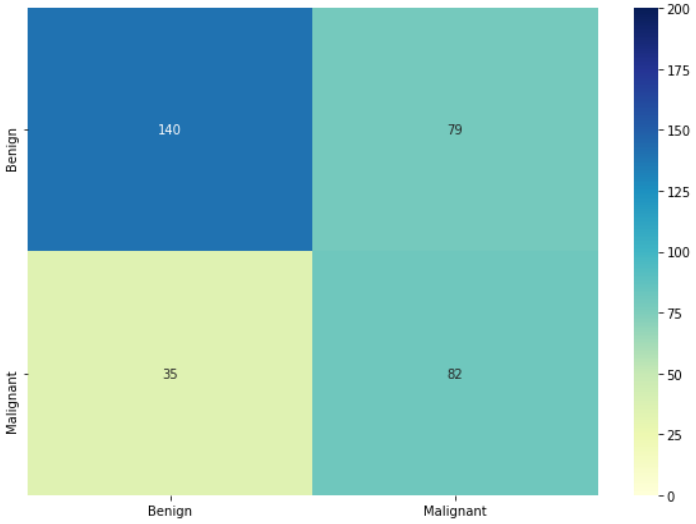

Figure 53: Confusion matrix computed on the test set using the experiment 12 model.

4.2.3 Results

To evaluate the best model, same observations made in “Benign”/”Malignant” classification for the scratch models are exploited.

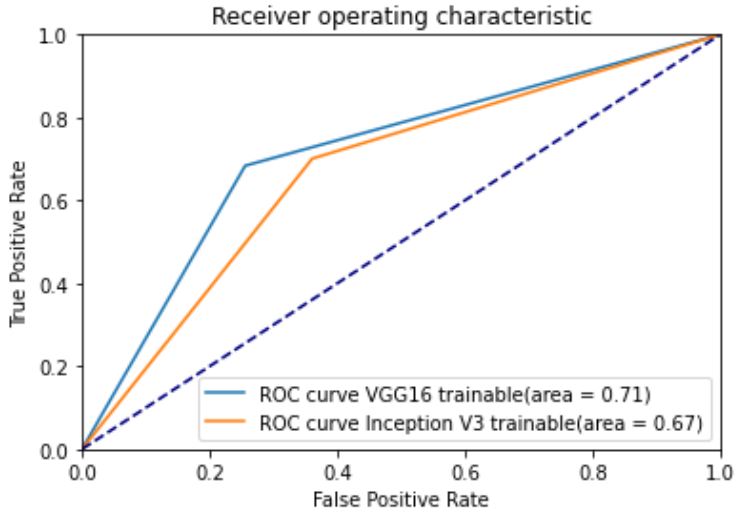

Figure 54: Comparison between the ROC curves computed on the four models based on "Benign"/"Malignant" classification.

	Recall Malignant	**F2-score**	**Epochs**
VGG16 trainable	0.68	**0.66**	25
Inception V3 trainable	**0.70**	0.65	13

Recall on the malignant class is higher for the trainable Inception V3 model, while trainable VGG16 reaches a higher F2-score.

Differences in both cases are not significant, so the two models can be considered equally valid in terms of performances.

5 EXPLOITING BASELINE PATCHES

5.1 MASS VS CALCIFICATION

5.1.1 Experiment 13: Siamese VGG16 with difference of features

In this experiment a Siamese architecture is employed. The architecture of the network is the following:

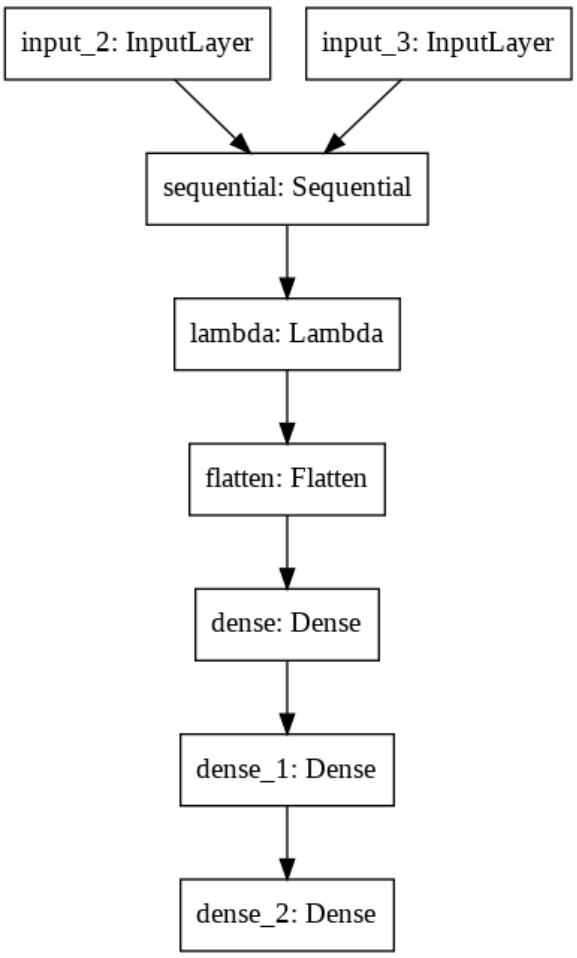

Figure 55: Architecture of the siamese learning approach, used in experiments 13 and 14.

In particular the Sequential layer is the VGG16, pretrained on ImageNet. The lambda layer performs a simple difference of the features extracted on the two inputs by the network. Finally, the two dense layers are composed by 512 and 256 neurons. The classification is performed using a binary cross entropy loss. The data augmentation and the class weighting is performed as in section 3.1.3 (DCCNN v1).

The best results of the model are obtained with:

- Epochs: 21
- Training accuracy: 64.72%
- Training loss: 0.6089
- Validation accuracy: 67.41%
- Validation loss: 0.5903
- Test accuracy: **66.37%**
- Test loss: 0.6109

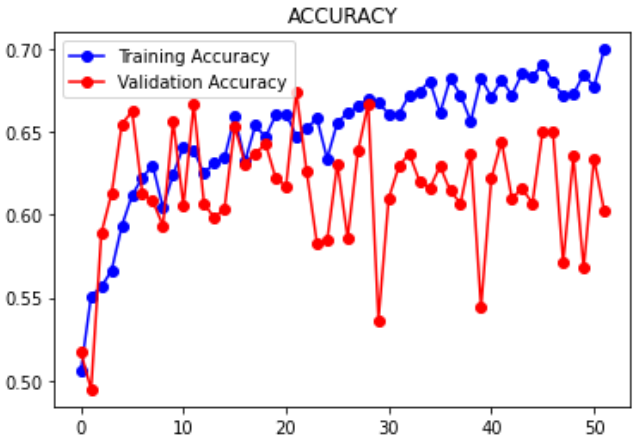

Figure 56: Comparison between the accuracy in function of the number of epochs on the training set and the validation set in experiment 13.

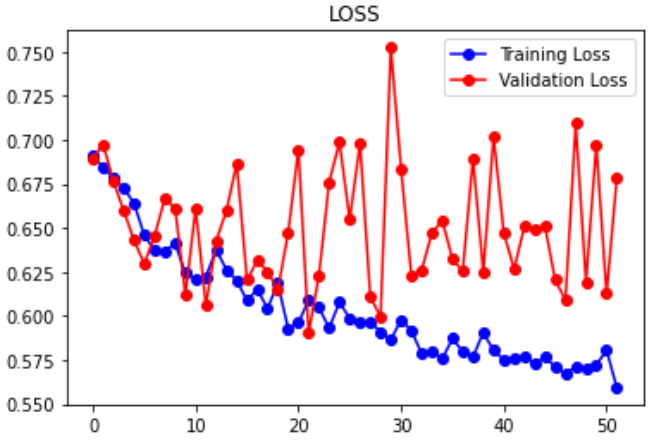

Figure 57: Comparison between the loss in function of the number of epochs on the training set and the validation set in experiment 13.

	Precision	**Recall**	**F1-score**
Mass	0.71	0.63	0.67
Calcification	0.62	0.70	0.66

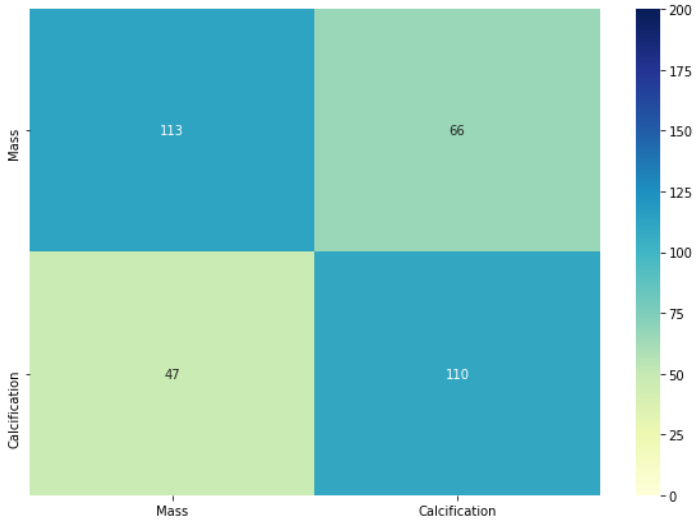

Figure 58: Confusion matrix computed on the test set using the experiment 13 model.

5.1.2 Experiment 14: Siamese VGG16 with deep-fusion

The architecture of the model used in this experiment is the same of the previous experiment. The only difference is within Lambda layer: instead of performing the difference between features, it performs the concatenation ("Multi-tasking Siamese Networks for Breast Mass Detection Using Dual-View Mammogram Matching", (Yuton, et al., 2020)).

The best results are obtained with:

- Epochs: 61
- Training accuracy: 69.50%
- Training loss: 0.5522
- Validation accuracy: 68.60%
- Validation loss: 0.5828
- Test accuracy: 65.18%
- Test loss: 0.5880

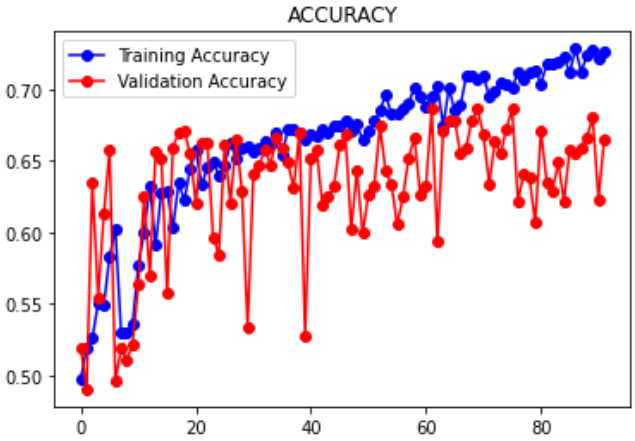

Figure 59: Comparison between the accuracy in function of the number of epochs on the training set and the validation set in experiment 14.

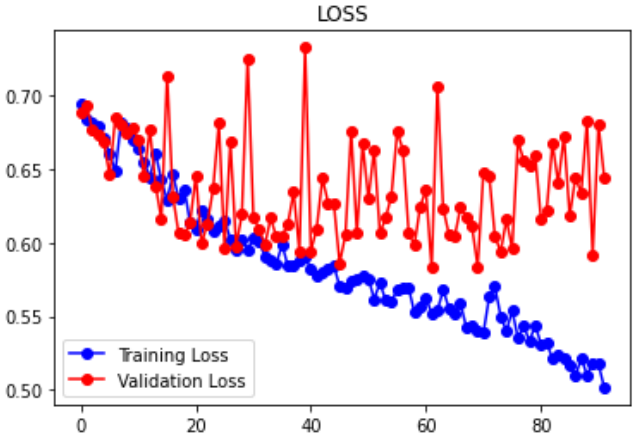

Figure 60: Comparison between the accuracy in function of the number of epochs on the training set and the validation set in experiment 14.

	Precision	**Recall**	**F1-score**
Mass	0.66	0.73	0.69
Calcification	0.64	0.57	0.60

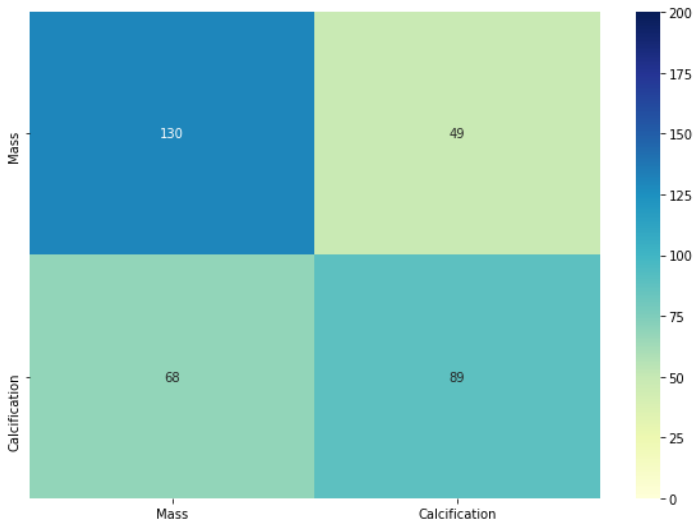

Figure 61: Confusion matrix computed on the test set using the experiment 14 model.

5.1.3 Experiment 15: Multi-modal architecture with difference of features

The idea of this model is taken from "Optimizing and Visualizing Deep Learning for Benign/Malignant Classification in Breast Tumors " (Yi, et al., 2017). In this paper, a multi-modal architecture is used:

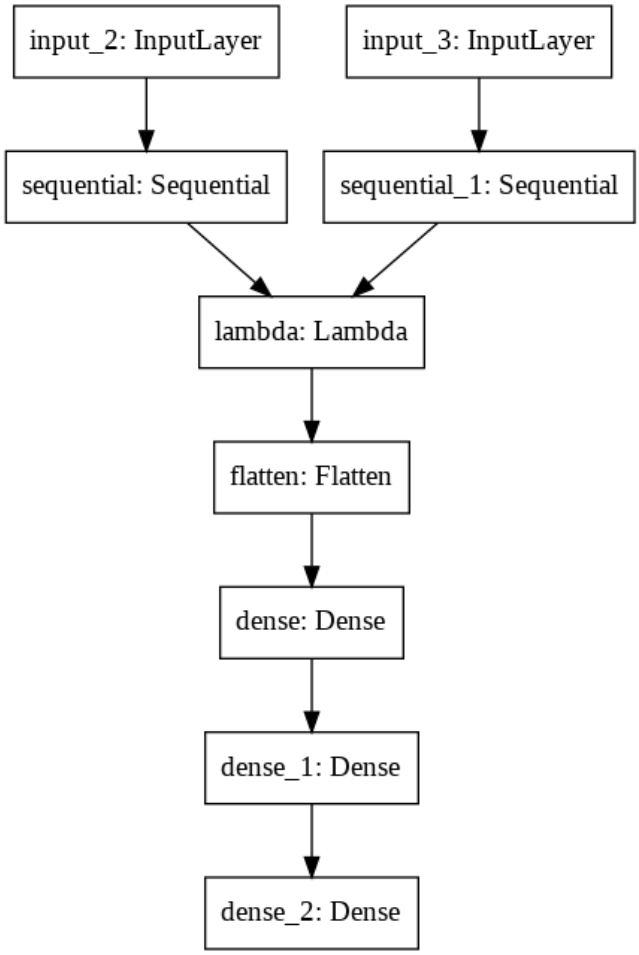

Figure 62: Architecture of the multi-modal learning approach, used in experiments 15 and 16.

It can be noticed that this model uses two Sequential layers, implementing two networks. One network gets as input the baseline patches and the other one takes the abnormality patches. Furthermore, a Lambda layer, performing the difference between features extracted from the two networks is employed. The two dense layers are composed by 512 and 256 neurons, respectively. The classification is performed using a binary cross entropy loss. The data augmentation and the class weighting are performed as in section 3.1.3 (DCCNN v1). The experiment consists of two sub-experiments.

In the first one, two identical networks in the Sequential layers are used and trained on Baseline and Abnormality patches, respectively. In particular, the network used in the Sequential layers is the VGG16, pre-trained on ImageNet.

In the second experiment, two different networks have been used: for the baseline patches, a VGG16 pre-trained on ImageNet is employed, while for the abnormality patches the model trained in experiment 4.1.2 (trainable VGG16) is loaded and used. Moreover, the VGG16 assigned on the baseline is trainable, so it can be fine-tuned on the baseline patches, while the model from the previous experiment is not fine-tuned, since it is already trained on abnormality patches.

The best results of the first sub-experiment (v1) are obtained with:

- Epochs: 18
- Training accuracy: 63.62%
- Training loss: 0.6262
- Validation accuracy: 68.75%
- Validation loss: 0.5944
- Test accuracy: **66.07%**
- Test loss: 0.6140

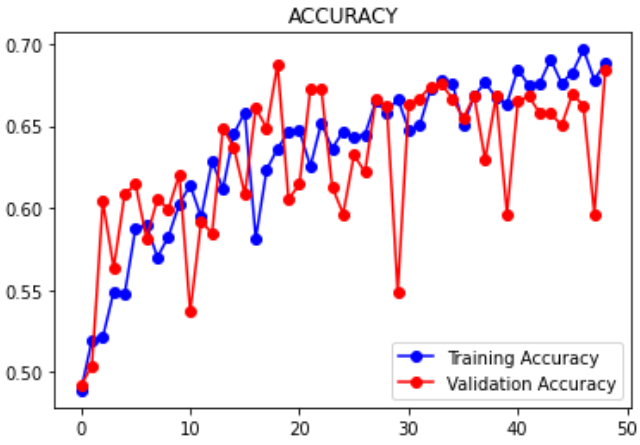

Figure 63: Comparison between the accuracy in function of the number of epochs on the training set and the validation set in the first model of experiment 15.

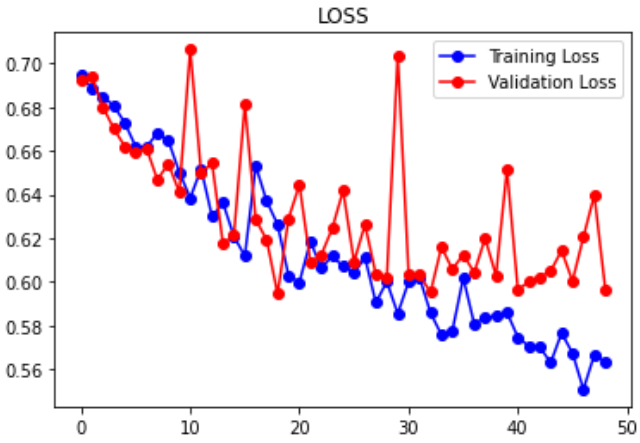

Figure 64: Comparison between the loss in function of the number of epochs on the training set and the validation set in the first model of experiment 15.

	Precision	**Recall**	**F1-score**
Mass	0.73	0.57	0.64
Calcification	0.61	0.76	0.68

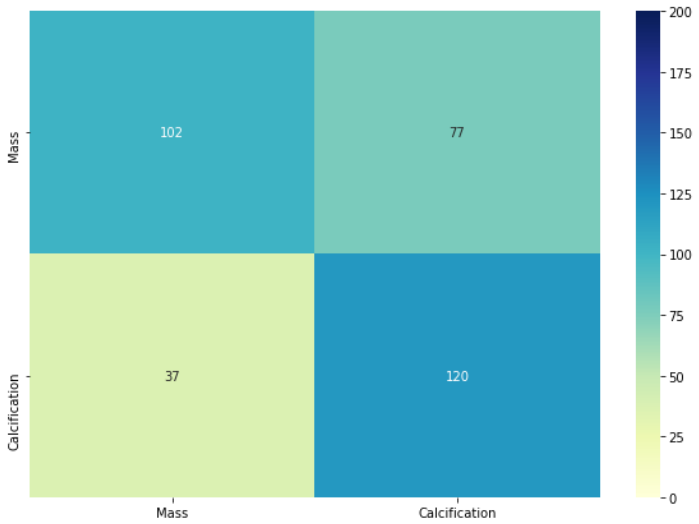

Figure 65: Confusion matrix computed on the test set using the first model of experiment 15.

The best results of the second sub-experiment (v2) are obtained with:

- Epochs: 48
- Training accuracy: 69.95%
- Training loss: 0.5545
- Validation accuracy: 69.05%
- Validation loss: 0.5719
- Test accuracy: **66.96%**
- Test loss: 0.6001

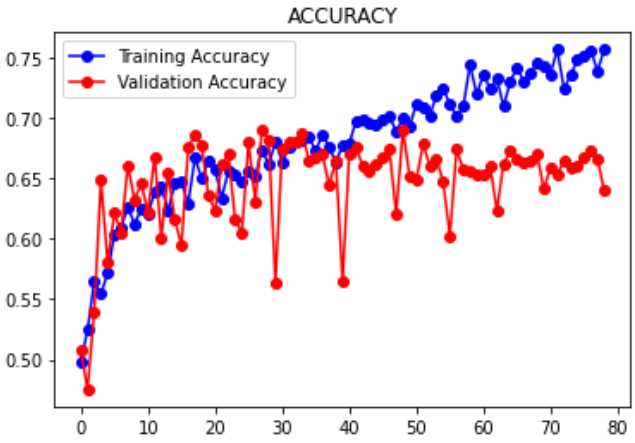

Figure 66: Comparison between the accuracy in function of the number of epochs on the training set and the validation set in the second model of experiment 15.

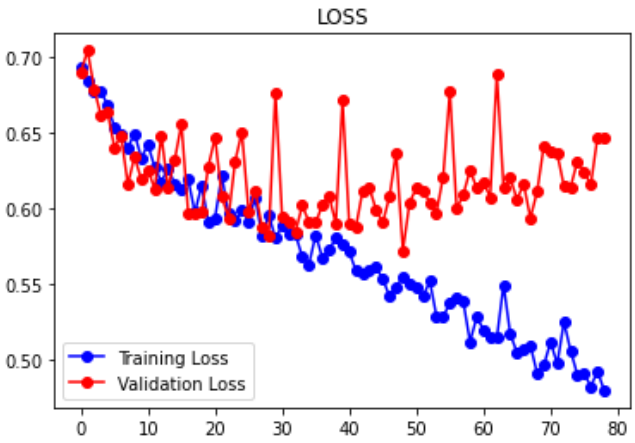

Figure 67: Comparison between the loss in function of the number of epochs on the training set and the validation set in the second model of experiment 15.

	Precision	**Recall**	**F1-score**
Mass	0.67	0.74	0.71
Calcification	0.67	0.59	0.62

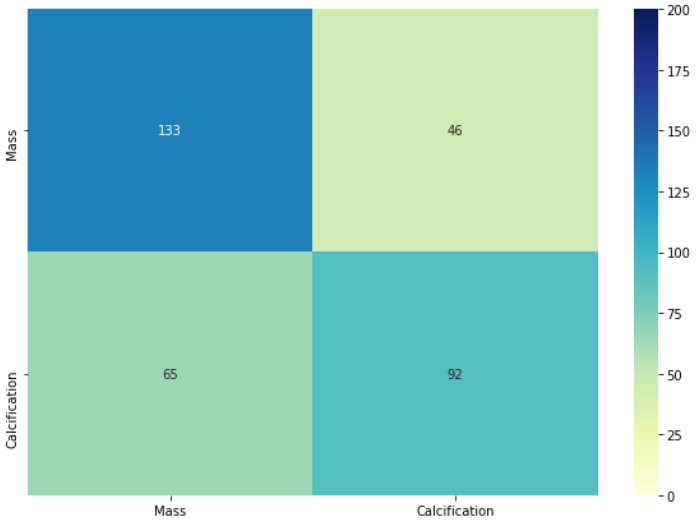

Figure 68: Confusion matrix computed on the test set using the second model of experiment 15.

5.1.4 Experiment 16: Multi-modal architecture with deep-fusion

The architecture of the model used in this experiment is the same of the previous experiment. The only difference is within Lambda layer: instead of performing the difference between features, it performs the concatenation.

Again, this experiment is split into two sub-experiments like before: first experiment uses two identical networks (VGG16 pre-trained on ImageNet), while second experiment uses two different networks, VGG16 and the model trained in experiment 4.1.2 (trainable VGG16).

The best results of the first sub-experiment (v3) are obtained with:

- Epochs: 67
- Training accuracy: 72.14%
- Training loss: 0.5357
- Validation accuracy: 67.26%
- Validation loss: 0.5981
- Test accuracy: **65.48%**
- Test loss: 0.6267

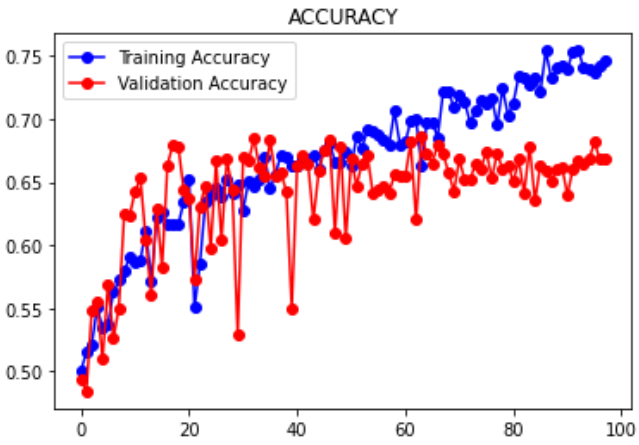

Figure 69: Comparison between the accuracy in function of the number of epochs on the training set and the validation set in the first model of experiment 16.

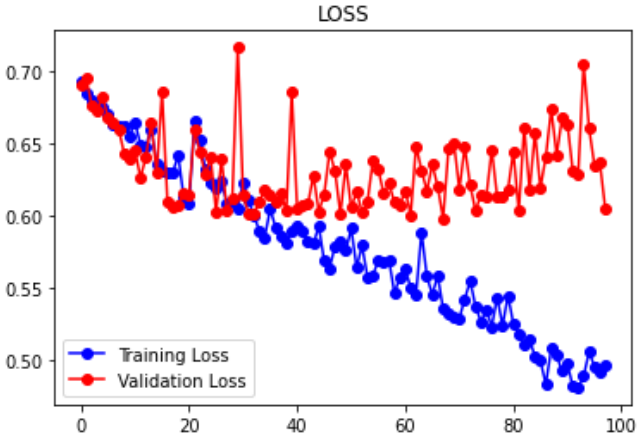

Figure 70: Comparison between the loss in function of the number of epochs on the training set and the validation set in the first model of experiment 16.

	Precision	**Recall**	**F1-score**

Mass	0.67	0.70	0.68
Calcification	0.64	0.61	0.62

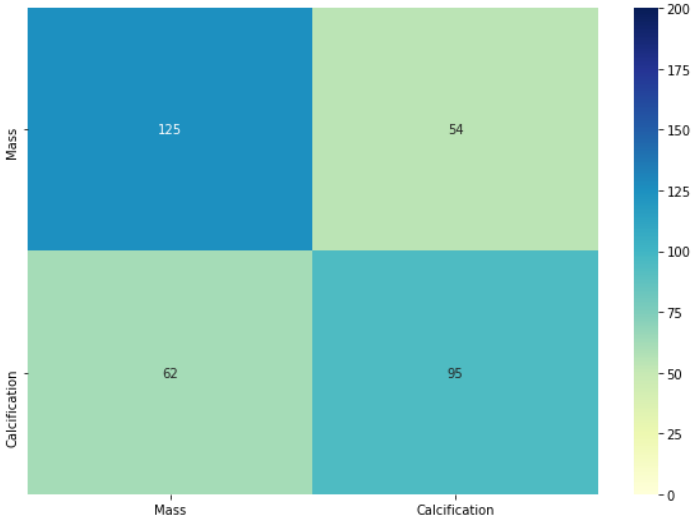

Figure 71: Confusion matrix computed on the test set using the first model of experiment 16.

The best results of the second sub-experiment (v4) are obtained with:

- Epochs: 28
- Training accuracy: 65.71%
- Training loss: 0.6003
- Validation accuracy: 68.75%
- Validation loss: 0.5910
- Test accuracy: **67.26%**
- Test loss: 0.6047

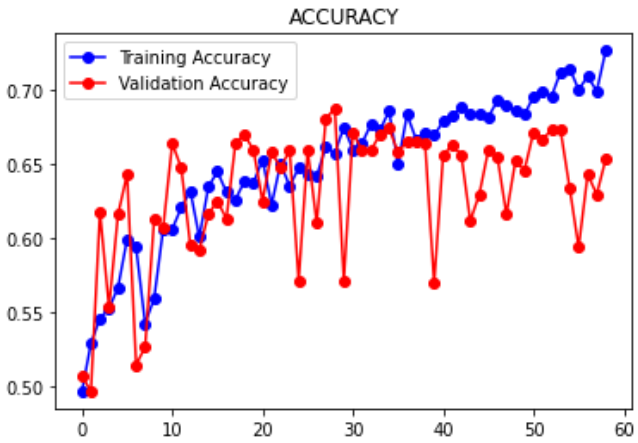

Figure 72: Comparison between the accuracy in function of the number of epochs on the training set and the validation set in the second model of experiment 16.

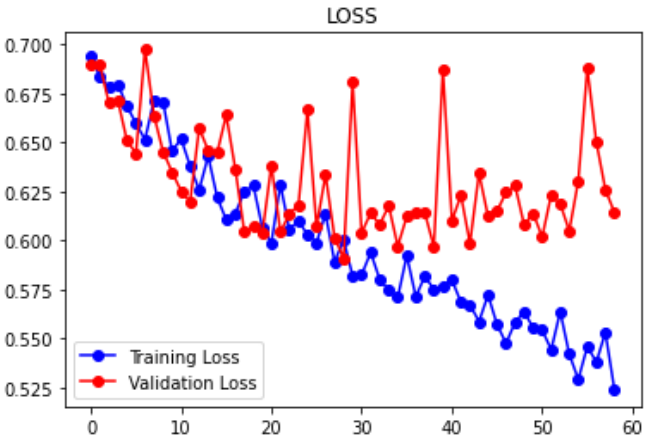

Figure 73: Comparison between the loss in function of the number of epochs on the training set and the validation set in the second model of experiment 16.

	Precision	**Recall**	**F1-score**
Mass	0.70	0.66	0.68
Calcification	0.64	0.68	0.66

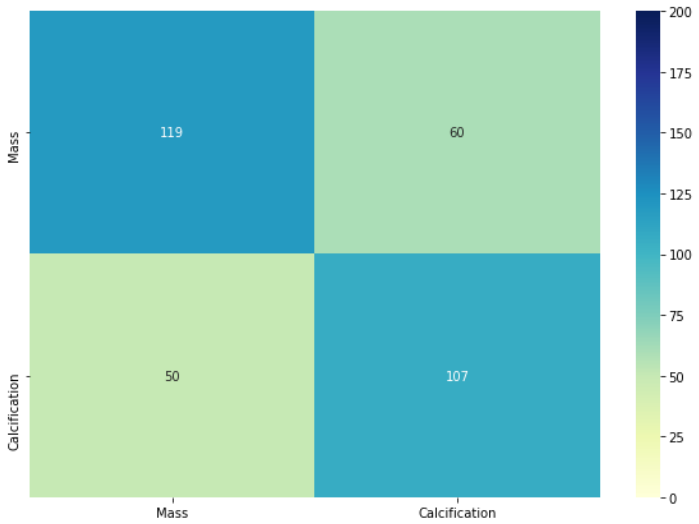

Figure 74: Confusion matrix computed on the test set using the second model of experiment 16.

5.1.5 Experiment 17: Concatenated Input

The idea behind this experiment is the following: what happens if the baseline patches are considered as an additional channel for the abnormality patches? Based on this question, the model developed allows to pass as input the abnormality patches and the baseline patches as single images with two channels, one corresponding to the abnormality and one corresponding to the baseline.

The architecture of the network employed is the same as in experiment 3.1.3 (DCCNN v1) since it is the best architecture that can accept a two-channel input. Even the training and the data augmentation are done in the same way.

The best results are obtained with:

- Epochs: 22
- Training accuracy: 70.98%

- Training loss: 1.8563
- Validation accuracy: 77.14%
- Validation loss: 1.1682
- Test accuracy: **76.19%**
- Test loss: 0.9505

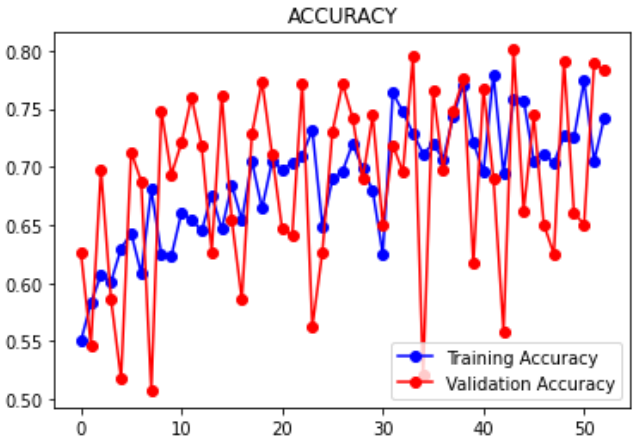

Figure 75: Comparison between the accuracy in function of the number of epochs on the training set and the validation set in experiment 17.

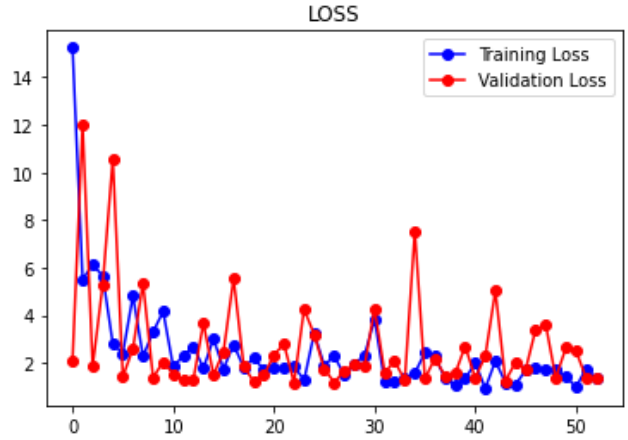

Figure 76: Comparison between the loss in function of the number of epochs on the training set and the validation set in experiment 17.

	Precision	**Recall**	**F1-score**
Mass	0.78	0.77	0.77
Calcification	0.74	0.76	0.75

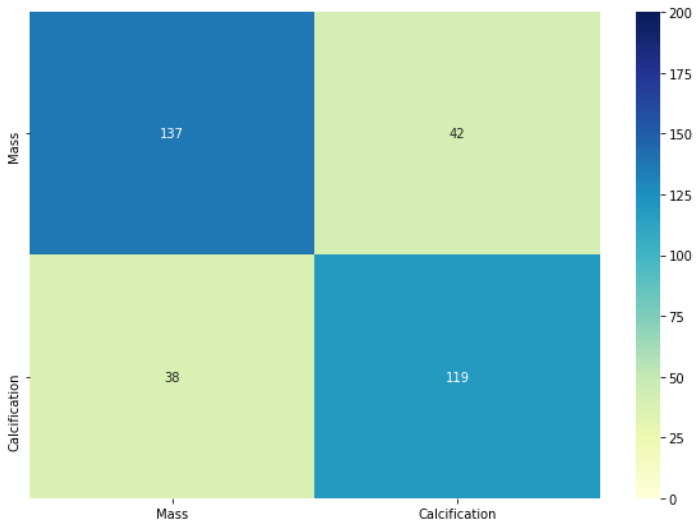

Figure 77: Confusion matrix computed on the test set using the experiment 17 model.

5.1.6 Results

As for the previous expertiments on “Mass”/”Calcification” classification, ROC curves and F1-scores are considered.

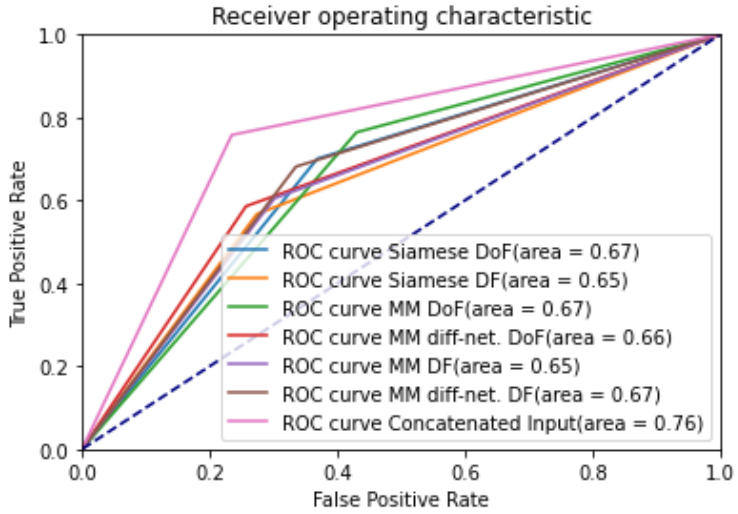

Figure 78: Comparison between the ROC curves computed on the four models based on "Mass"/"Calcification" classification. DoF: Difference of Features. DF: Deep-fusion. MM: Multi-modal. Diff-net: difference networks.

	Accuracy Test	**F1-Score**	**Epochs**
Siamese DoF	0.6637	0.66	21
Siamese DF	0.6518	0.65	61
MM DoF	0.6607	0.66	18
MM diff-net. DoF	0.6696	0.66	48
MM DF	0.6548	0.65	67
MM diff-net. DF	0.6726	0.67	28
Concatenated Input	**0.7619**	**0.76**	22

All the results are not good as the other approaches (namely scratch models and pre-trained models). Anyway, there is one best model among the others: the Concatenated Input approach outperforms significantly the other models in both F1-score and AUC.

The reason behind these results probably is that baseline patches do not add useful additional information in respect of the original abnormality patches and moreover they can fool the models.

5.2 BENIGN VS MALIGNANT

In this section, the best architectures of the previous section are used to perform "Benign"/"Malignant" classification. In particular three architectures have been evaluated: the best one from the multimodal architecture, the concatenated input architecture and the best one from the Siamese architectures.

5.2.1 Experiment 18: Concatenated Input

This experiment exploits the architecture used in 5.1.5, changing the classification task from "Mass"/"Calcification" to "Benign"/"Malignant".

The best results are obtained with:

- Epochs: 18
- Training accuracy: 57.98%
- Training loss: 1.2127
- Validation accuracy: 57.61%
- Validation loss: 0.8411
- Test accuracy: **56.25%**
- Test loss: 0.9330

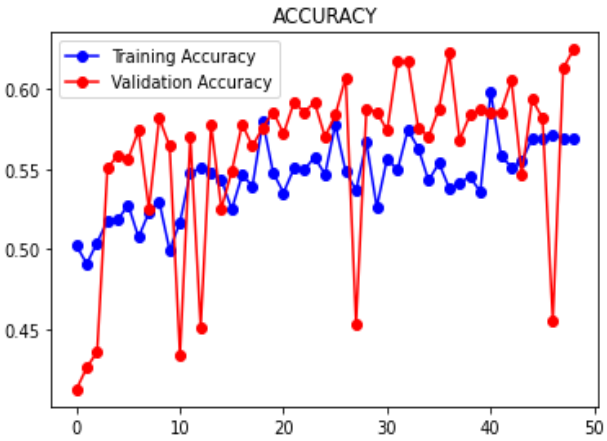

Figure 79: Comparison between the accuracy in function of the number of epochs on the training set and the validation set in experiment 18.

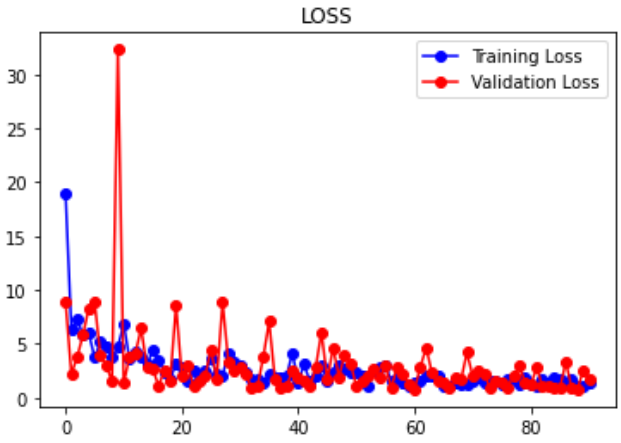

Figure 80: Comparison between the loss in function of the number of epochs on the training set and the validation set in experiment 18.

	Precision	**Recall**	**F1-score**
Benign	0.72	0.54	0.62
Malignant	0.41	0.60	0.49

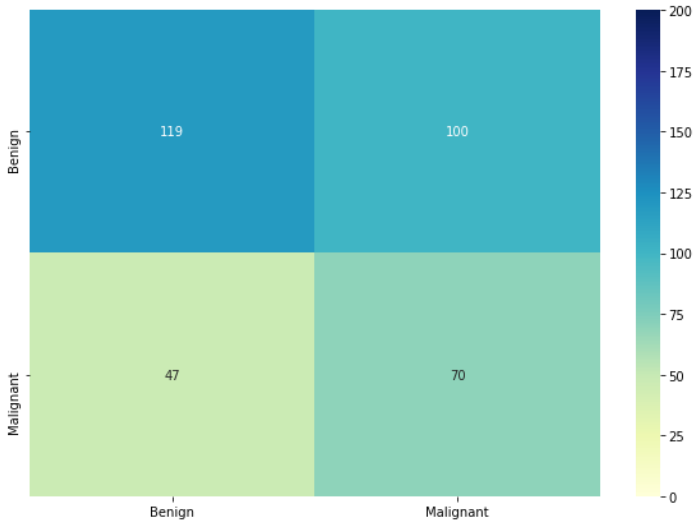

Figure 81: Confusion matrix computed on the test set using the experiment 18 model.

5.2.2 Experiment 19: Best multi-modal architecture

The architecture chosen is the one from the second sub-experiment of 5.1.4: the multi-modal architecture that uses two different networks with deep fusion.

The best results are obtained with:

- Epochs: 48
- Training accuracy: 63.92%
- Training loss: 0.6230
- Validation accuracy: 64.72%
- Validation loss: 0.6249
- Test accuracy: **61.90%**
- Test loss: 0.6920

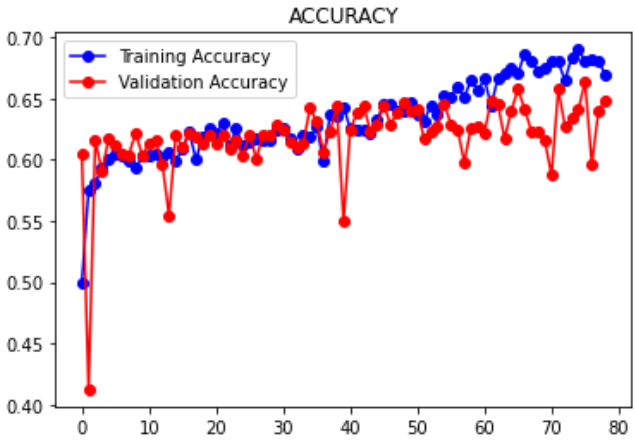

Figure 82: Comparison between the accuracy in function of the number of epochs on the training set and the validation set in experiment 19.

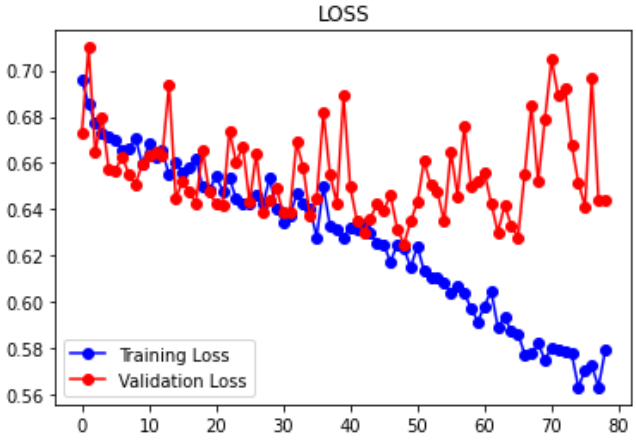

Figure 83: Comparison between the loss in function of the number of epochs on the training set and the validation set in experiment 19.

	Precision	**Recall**	**F1-score**
Benign	0.68	0.79	0.73
Malignant	0.43	0.31	0.36

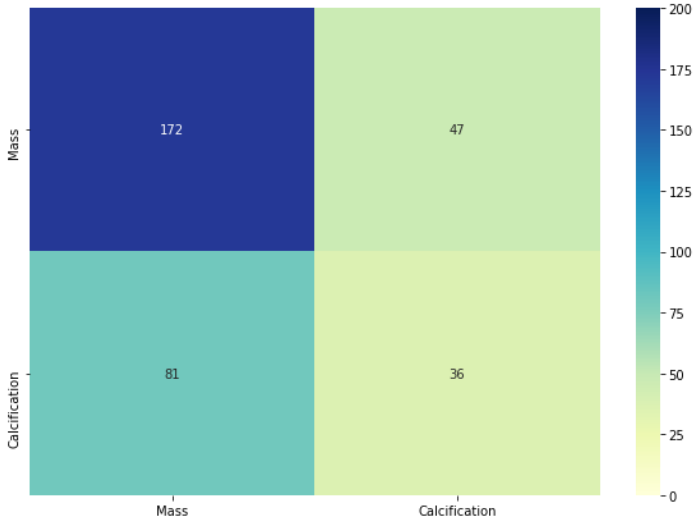

Figure 84: Confusion matrix computed on the test set using the experiment 19 model.

5.2.3 Experiment 20: Best Siamese architecture

The architecture chosen is the Siamese architecture that uses difference of features (section 5.1.1).

The best results are obtained with:

- Epochs: 48
- Training accuracy: 63.07%
- Training loss: 0.6282
- Validation accuracy: 64.72%
- Validation loss: 0.6339
- Test accuracy: **58.63%**
- Test loss: 0.7039

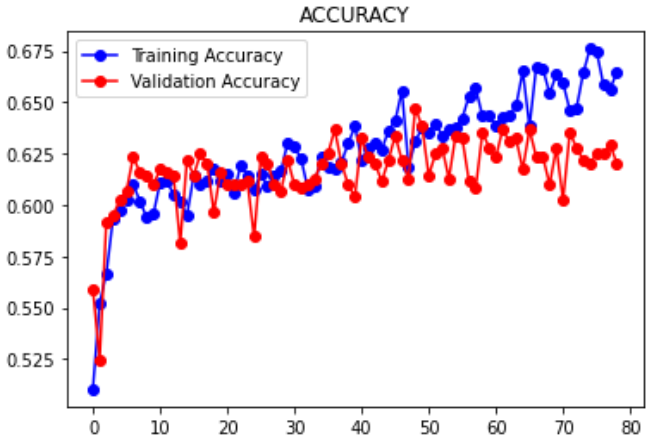

Figure 85: Comparison between the accuracy in function of the number of epochs on the training set and the validation set in experiment 20.

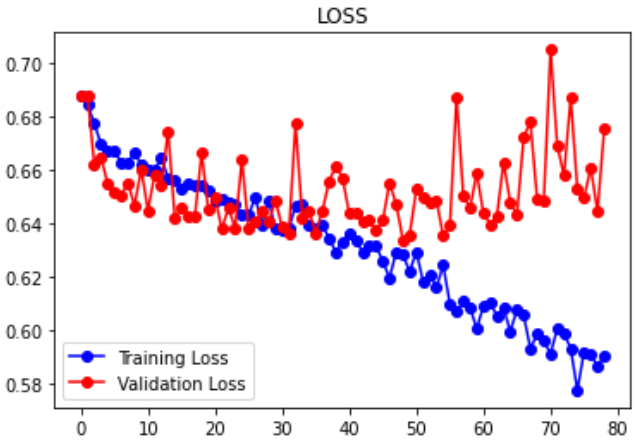

Figure 86: Comparison between the loss in function of the number of epochs on the training set and the validation set in experiment 20.

	Precision	**Recall**	**F1-score**
Benign	0.70	0.65	0.67
Malignant	0.42	0.47	0.44

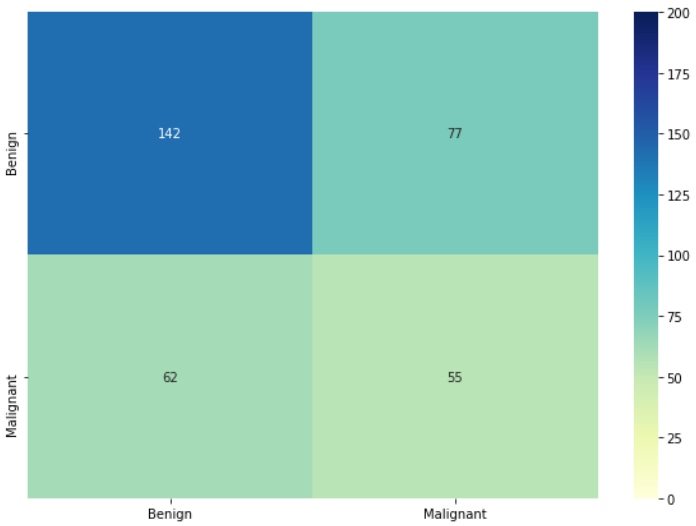

Figure 87: Confusion matrix computed on the test set using the experiment 20 model.

5.2.4 Results

To evaluate the best model, according to the observations done in the previous "Benign"/"Malignant" classification task, the recall on the malignant class and the F2-score are considered.

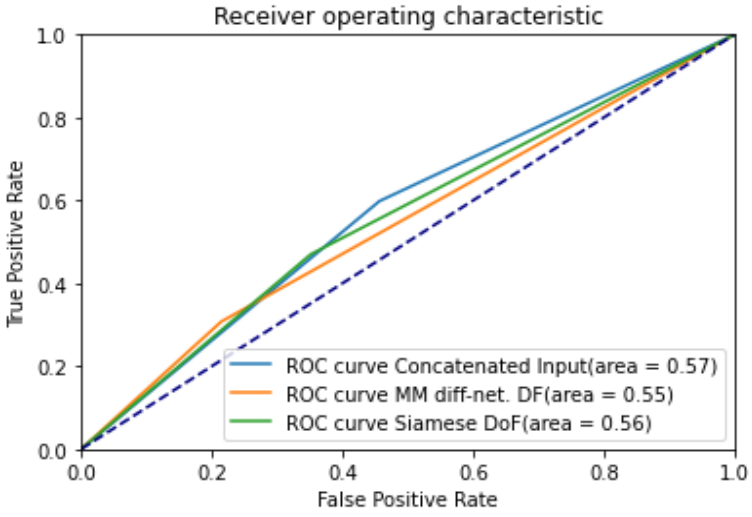

Figure 88: Comparison between the ROC curves computed on the four models based on "Benign"/"Malignant" classification.

	Recall Malignant	**F2-score**	**Epochs**
Concatenated Input	**0.60**	**0.5485**	18
MM diff-net. DF	0.31	0.3266	48
Siamese DoF	0.47	0.4583	48

Even in this case the Concatenated Input outperforms the other models in terms of recall on malignant class and F2-score. The results obtained cannot be considered satisfying. The baseline patches, as for "Mass"/"Calcification", have decreased the final performance of the models.

6 MODEL ENSEMBLING

6.1 MASS VS CALCIFICATION

In this part, model ensembling techniques are used in order to improve performances. In particular, three techniques have been considered:

1. **Majority voting**: the assigned class to the sample is the class that receives the majority of votes from the predictions made by the base models.
2. **Averaging**: the output score of the classification is a weighted sum of the probabilities given to the sample by the base models. The weights are assigned according to the validation accuracy of the models. The assigned class is determined by the rounding of the output score.
3. **Model stacking**: a new model is built on the outputs coming from the base models. This new model learns how to correctly classify and interpret the probabilities provided by the base models.

For each of these approaches, the used base models are: DCCNN v1 model (3.1.3), VGG16 trainable (4.1.2) and Inception V3 trainable (4.1.4). These three models are different from each other in their architectures, thus introducing diversity among base classifiers.

6.1.1 Majority Voting

Accuracy test set: **90.77%**

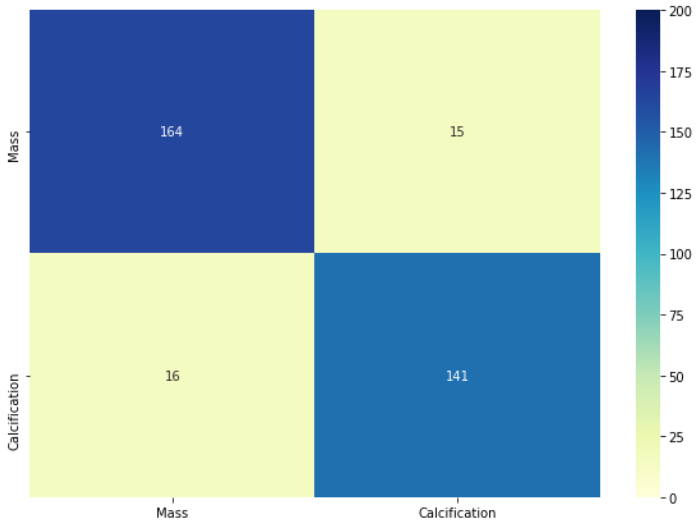

Figure 89: Confusion matrix computed on the test set using the experiment on the majority voting.

	Precision	**Recall**	**F1-score**
Mass	0.91	0.92	0.91
Calcification	0.90	0.90	0.90

6.1.2 Averaging

Accuracy test set: **91.07%**

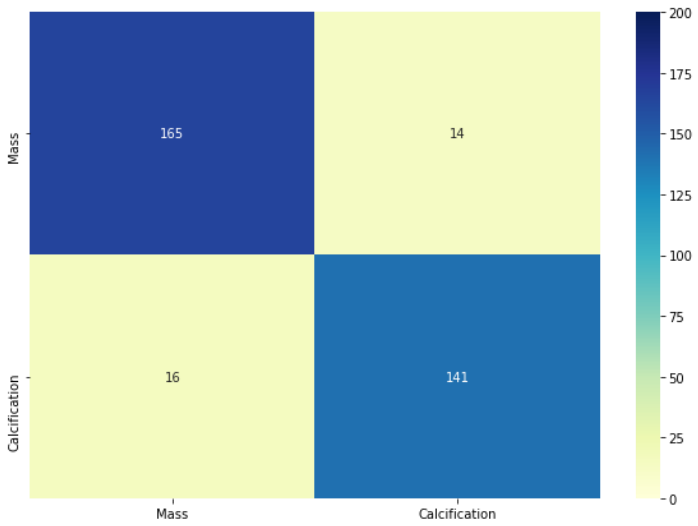

Figure 90: Confusion matrix computed on the test set using the experiment on the averaging.

	Precision	**Recall**	**F1-score**
Mass	0.91	0.92	0.92
Calcification	0.91	0.90	0.90

6.1.3 Model Stacking with Neural Network as Meta-Classifier

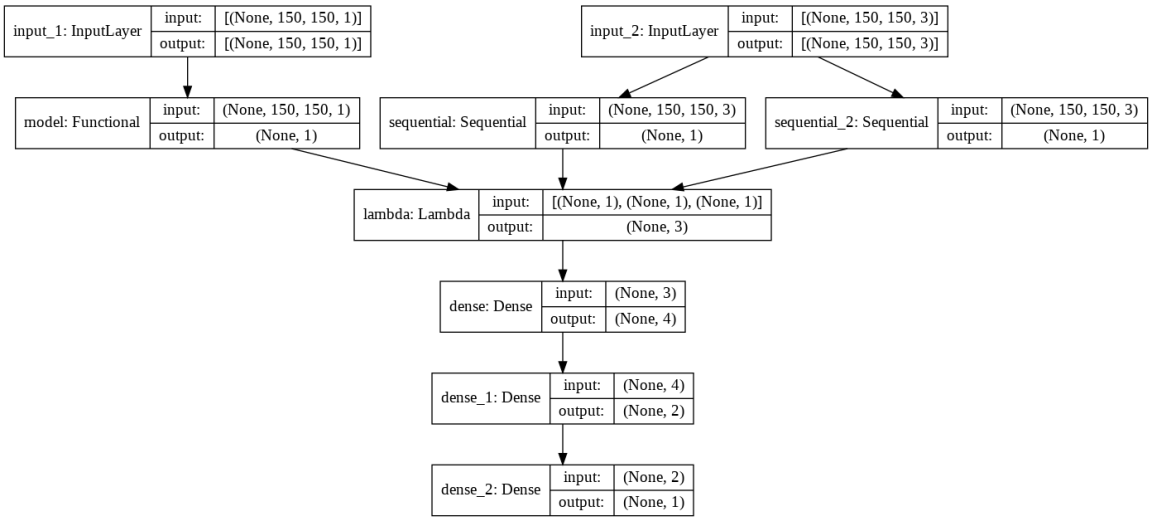

Figure 91: Architecture of the model stacking approach using DCCNN, VGG16 trainable and Inception V3 trainable as base models.

The best results are obtained with:

- Epochs: 31
- Training accuracy: 96.81%
- Training loss: 0.0841
- Validation accuracy: 97.61%
- Validation loss: 0.0904
- Test accuracy: **91.07%**
- Test loss: 0.2873

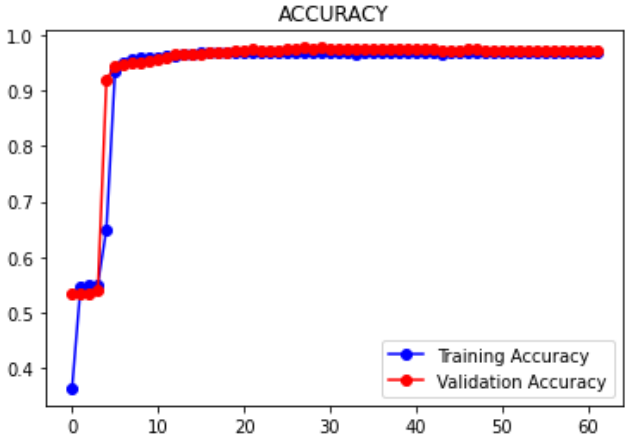

Figure 92: Comparison between the accuracy in function of the number of epochs on the training set and the validation set in experiment on model stacking.

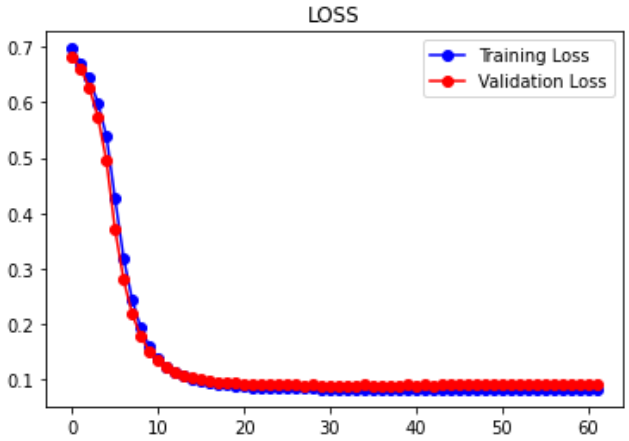

Figure 93: Comparison between the loss in function of the number of epochs on the training set and the validation set in experiment on model stacking.

	Precision	**Recall**	**F1-score**
Mass	0.93	0.91	0.92
Calcification	0.89	0.92	0.91

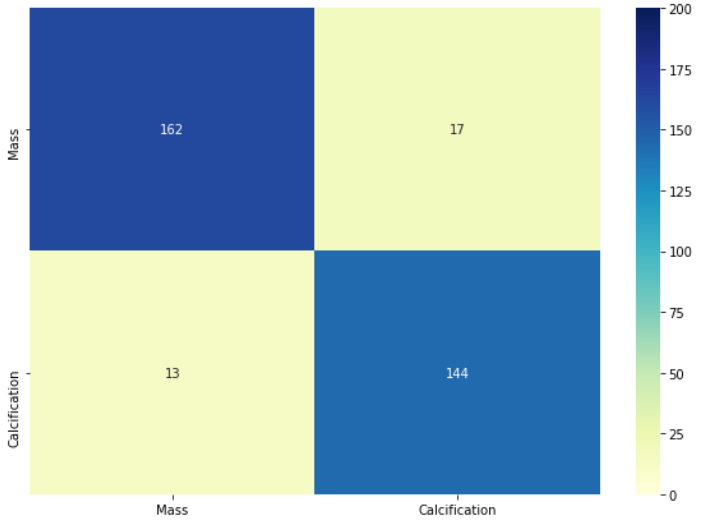

Figure 94: Confusion matrix computed on the test set using the experiment on the model stacking with Neural Network as Meta-Classifier.

6.1.4 Model Stacking with Non-Deep Learning classifiers as Meta-Classifiers

In this section several Non-Deep Learning classifiers are exploited as meta-classifiers for the model stacking approach:

1. **SVM**

	Precision	**Recall**	**F1-score**
Mass	0.92	0.91	0.91
Calcification	0.90	0.90	0.90

   a. Test accuracy: 90.77%
   b. F1-score: 90.74%
   c. AUC: 0.91

2. **Gaussian Naïve Bayesian**

	Precision	**Recall**	**F1-score**
Mass	0.93	0.91	0.92
Calcification	0.90	0.92	0.91

   a. Test accuracy: 91.37%
   b. F1-score: 91.34%
   c. AUC: 0.91

3. **K-NN Classifier (K=3)**

	Precision	**Recall**	**F1-score**
Mass	0.92	0.91	0.91
Calcification	0.89	0.90	0.90

   a. Test accuracy: 90.48%
   b. F1-score: 90.44%
   c. AUC: 0.90

4. **Random Forest**

	Precision	**Recall**	**F1-score**
Mass	0.92	0.91	0.91
Calcification	0.89	0.91	0.90

a. Test accuracy: 90.77%
b. F1-score: 90.74%
c. AUC: 0.91

5. AdaBoost

	Precision	**Recall**	**F1-score**
Mass	0.93	0.91	0.92
Calcification	0.89	0.92	0.91

a. Test accuracy: 91.07%
b. F1-score: 91.04%
c. AUC: 0.91

The best model is the one that uses the Gaussian Naïve Bayesian classifier according to both F1-score (91.34%) and AUC (0.91). Hence, in the following only this model has been considered.

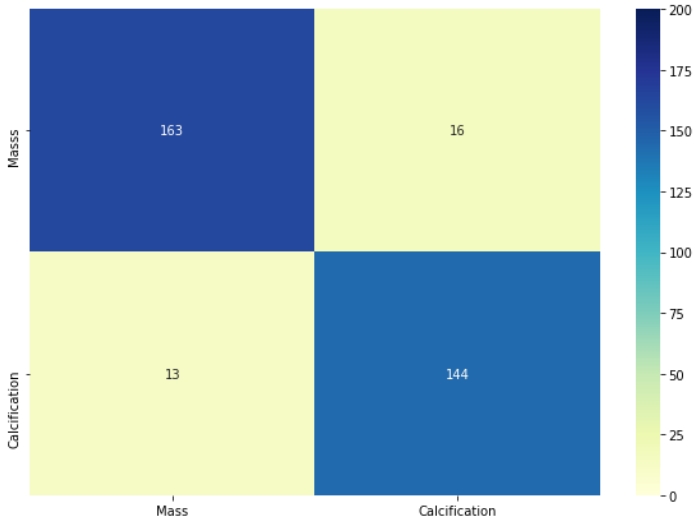

Figure 95: Confusion matrix computed on the test set using the experiment on the model stacking with Gaussian Naïve Bayesian as Meta-Classifier.

6.1.5 Results

As for the "normal" models, also for the three ensembling techniques the F1-score and the AUC are considered for the evaluation of the performance on the "Mass"/"Calcification" classification task.

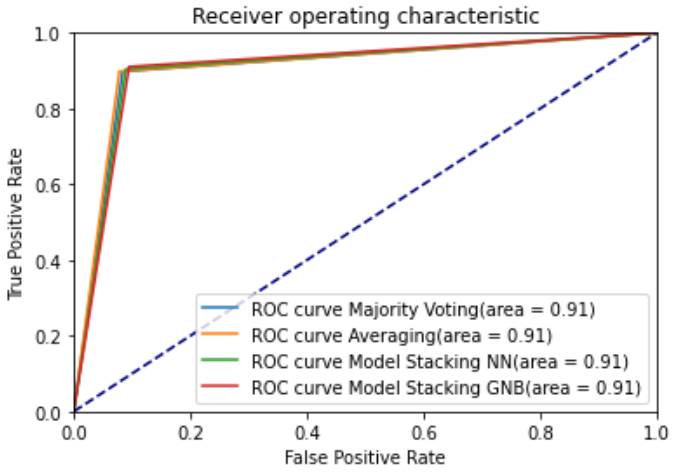

Figure 96: Comparison between the ROC curves computed on the four models based on "Mass"/"Calcification" classification.

	Accuracy Test	**F1-Score**	**AUC**
Majority Voting	0.9077	0.9073	0.9071
Averaging	0.9107	0.9102	0.9099
Model Stacking with NN	0.9107	0.9104	0.9111
Model Stacking with Gaussian NB	**0.9137**	**0.9134**	**0.9139**

The best model is the stacking model that uses the Gaussian Naïve Bayesian as meta-classifier. It outperforms the other models in both F1-score and AUC. The difference is not significant but it can be considered the best model among the ensembling solutions.

6.1.6 Comparison with respect to base classifiers

	Accuracy Test	**F1-Score**	**AUC**
DCCNN v1	0.8869	0.89	0.88
VGG16 trainable	0.8958	0.90	0.90
Inception V3 trainable	0.8929	0.89	0.89
Model Stacking with Gaussian NB	**0.9137**	**0.91**	**0.91**

As expected, the results obtained by the composite classifier outperform those obtained with the base classifiers. This is due to the fact that the model stacking finds the best way to combine their output probabilities. Assume that there is a sample that one of the base networks does not recognize; it is very likely that the output probability for that network and that sample is not prominent. If there is one of the other two networks that knows how to correctly classify the sample, its output probability will be prominent and so it will adjust the composite output probability covering the first network misclassification. This phenomenon is less present in the majority voting approach, in which only the prediction labels are used for the combined classification.

6.2 BENIGN VS MALIGNANT

6.2.1 Majority Voting

Accuracy test set: **69.64%**

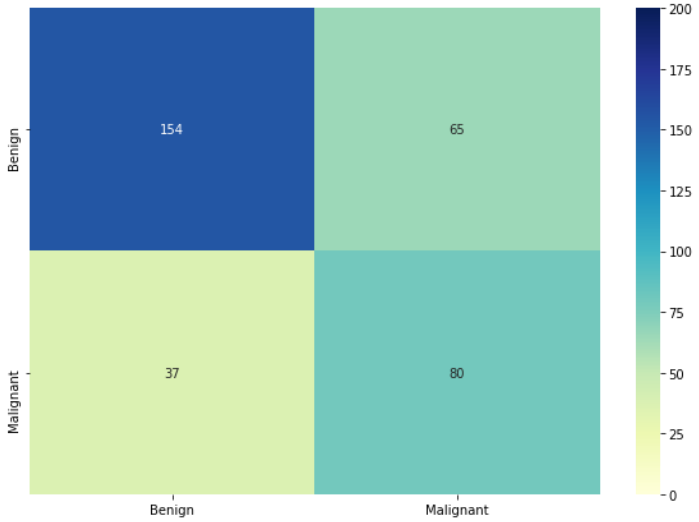

Figure 97: Confusion matrix computed on the test set using the experiment on the majority voting.

	Precision	**Recall**	**F1-score**
Benign	0.81	0.70	0.75
Malignant	0.55	0.68	0.61

6.2.2 Averaging

Accuracy test set: **71.43%**

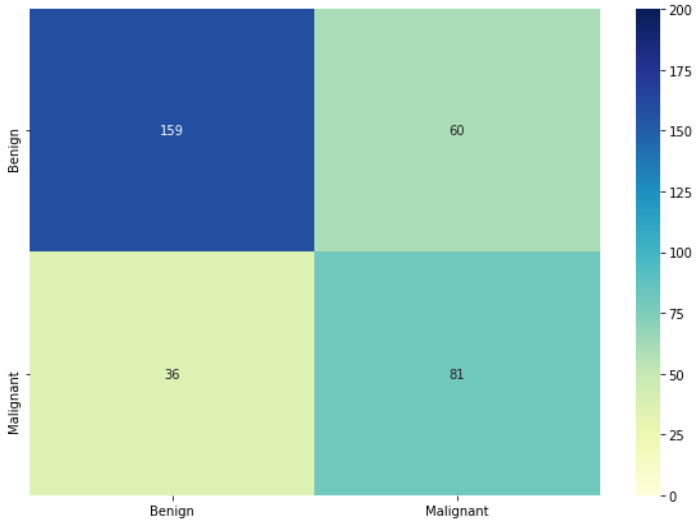

Figure 98: Confusion matrix computed on the test set using the experiment on the averaging.

	Precision	**Recall**	**F1-score**
Benign	0.82	0.73	0.77
Malignant	0.57	0.69	0.63

6.2.3 Model Stacking with Neural Network as Meta-Classifier

The best results are obtained with:

- Epochs: 249
- Training accuracy: 77.48%
- Training loss: 0.4474
- Validation accuracy: 79.67%
- Validation loss: 0.4382
- Test accuracy: **75.90%**
- Test loss: 0.5743

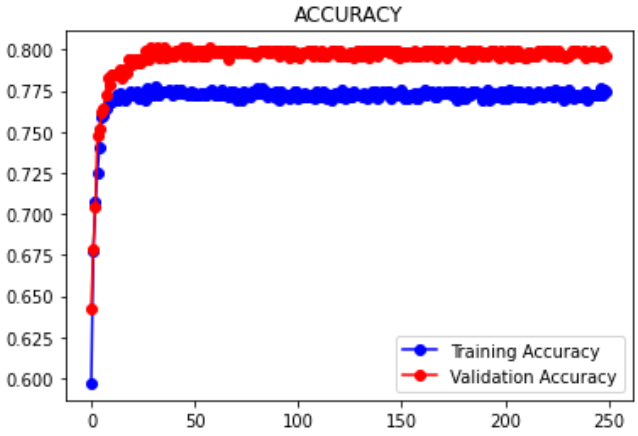

Figure 99: Comparison between the accuracy in function of the number of epochs on the training set and the validation set in experiment on model stacking.

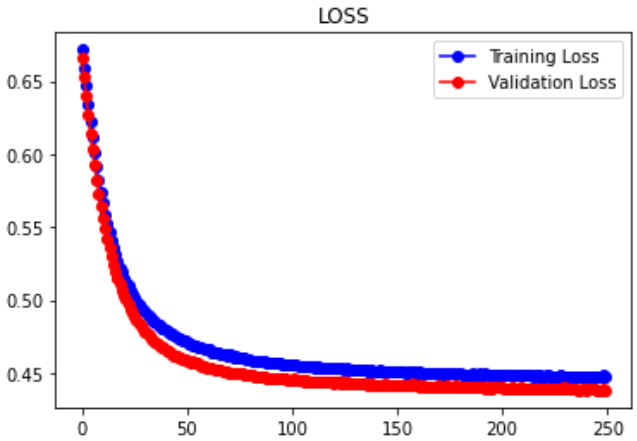

Figure 100: Comparison between the loss in function of the number of epochs on the training set and the validation set in experiment on model stacking.

	Precision	**Recall**	**F1-score**
Benign	0.84	0.78	0.81
Malignant	0.63	0.73	0.68

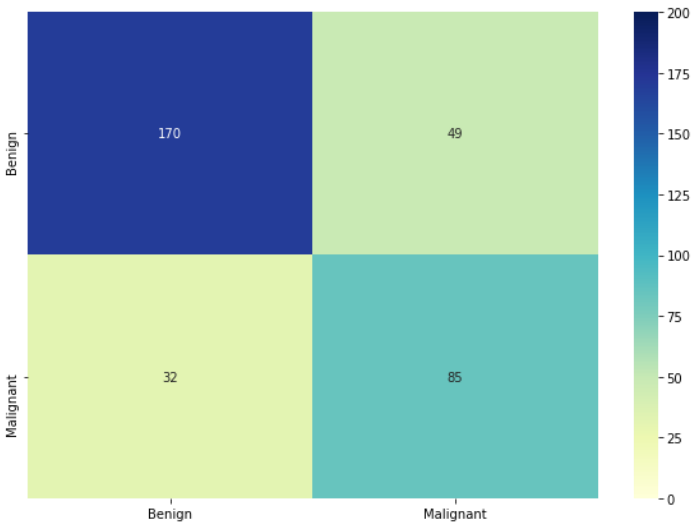

Figure 101: Confusion matrix computed on the test set using the experiment on the model stacking with the Neural Network as Meta-Classifier.

6.2.4 Model Stacking with Non-Deep Learning classifiers as Meta-Classifiers

As it has been done for "Mass"/"Calcification" classification, in this section several Non-Deep Learning classifiers are exploited as meta-classifiers for the model stacking approach:

1. **SVM**

	Precision	**Recall**	**F1-score**
Benign	0.84	0.77	0.80
Malignant	0.63	0.73	0.67

   a. Test accuracy: 75.59%
   b. F2-score: 70.48%
   c. AUC: 0.75

2. **Gaussian Naïve Bayesian**

	Precision	**Recall**	**F1-score**
Benign	0.83	0.68	0.75
Malignant	0.55	0.74	0.63

   a. Test accuracy: 70.24%
   b. F2-score: 69.02%
   c. AUC: 0.71

3. **K-NN Classifier (K=3)**

	Precision	**Recall**	**F1-score**
Benign	0.80	0.68	0.73
Malignant	0.53	0.69	0.60

   a. Test accuracy: 68.15%
   b. F2-score: 65.32%
   c. AUC: 0.68

4. **Random Forest**

	Precision	**Recall**	**F1-score**
Benign	0.80	0.69	0.74
Malignant	0.54	0.68	0.60

   a. Test accuracy: 68.45%

b. F2-score: 64.23%
c. AUC: 0.68

5. **AdaBoost**

	Precision	**Recall**	**F1-score**
Benign	0.82	0.74	0.78
Malignant	0.59	0.70	0.64

a. Test accuracy: 72.92%
b. F2-score: 67.66%
c. AUC: 0.72

The best model according to the recall on the malignant is the Gaussian Naïve Bayesian, with a value equal to 0.74, but according to the F2-score the SVM is the best one with a value equal to 70.48%. To decide among them, it is possible to consider also the accuracy and the AUC. In both of them, the SVM outperforms the Gaussian Naïve Bayesian, and for that reason it can be considered the best model.

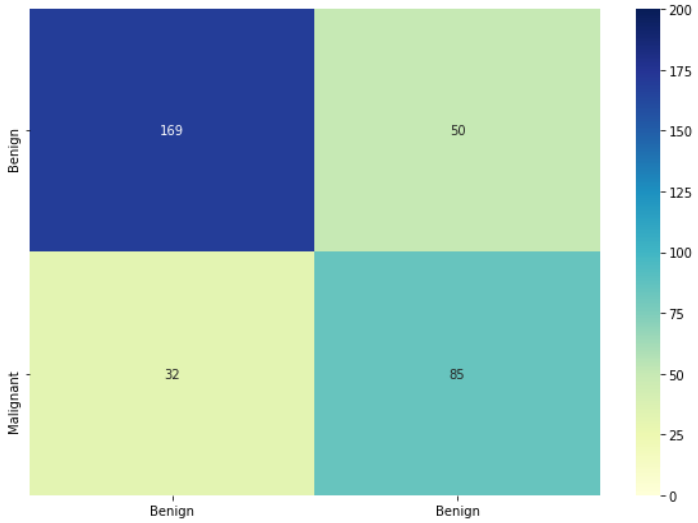

Figure 102: Confusion matrix computed on the test set using the experiment on the model stacking with the SVM as Meta-Classifier.

6.2.5 Results

As for the other "Benign"/"Malignant" classification tasks, F2-score and recall on malignant are considered.

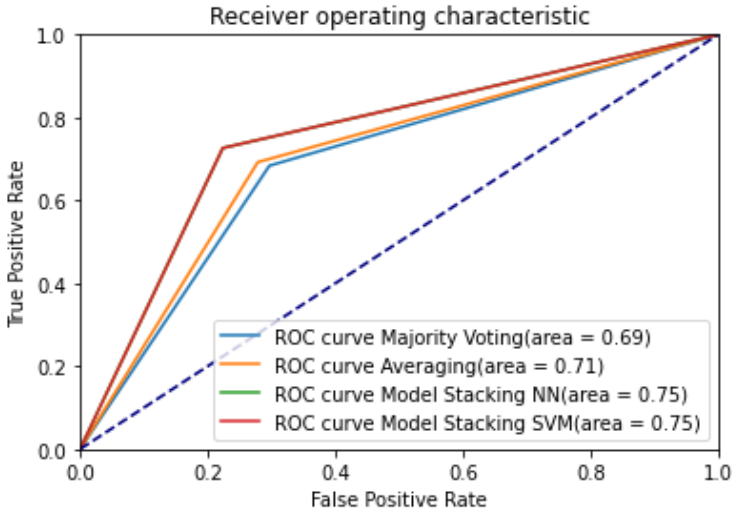

Figure 103: Comparison between the ROC curves computed on the four models based on "Benign"/"Malignant" classification.

	Recall Malignant	**F2-score**	**AUC**
Majority Voting	0.6838	0.6525	0.6935
Averaging	0.6923	0.6650	0.7092
Model Stacking with NN	**0.7265**	**0.7060**	**0.7514**
Model Stacking with SVM	**0.7265**	0.7048	0.7491

According to the F2-score and the recall on the "Malignant" class, the model stacking that uses the Neural Network as meta-classifier is the best model because it outperforms the others and it has the same recall of the model stacking with SVM. Moreover, it outperforms the others on the AUC.

6.2.6 Comparison with respect to base classifiers

	Recall Malignant	**F2-score**	**AUC**
DCCNN v1	0.62	0.59	0.64
VGG16 trainable	0.68	0.66	0.71
Inception V3	0.70	0.65	0.67
Model Stacking with NN	**0.73**	**0.71**	**0.75**

For the "Benign"/"Malignant" classification task, in which the performances of the base classifiers are worse than their corresponding models in the "Mass"/"Calcification" classification task, the improvement given by the model stacking approach is more prominent. This can be observed in both recall on malignant and F2-score, that are improved in respect of their highest value in the base classifiers.

7 FINAL RESULTS

7.1 MASS VS CALCIFICATION

In the following table the best models from each section are compared:

	Accuracy Test	F1-score	AUC
DCCNN v1	88.69%	0.89	0.88
VGG16 trainable	89.58%	0.90	0.90
Inception V3 trainable	89.29%	0.89	0.89
Concatenated Input	76.19%	0.76	0.76
Model Stacking with Gaussian NB	**91.37%**	**0.91**	**0.91**

The main criteria for choosing the best model are the F1-score and the AUC. According to these metrics, the model stacking that uses the Gaussian Naïve Bayesian as meta-classifier is the best one and for that reason it is the model used for the final submission.

7.2 BENIGN VS MALIGNANT

	Accuracy Test	Recall Malignant	F2-score	AUC
DCCNN v1	64.58%	0.62	0.59	0.64
VGG16 trainable	72.32%	0.68	0.66	0.71
Inception V3 trainable	66.07%	0.70	0.65	0.67
Concatenated Input	56.25%	0.60	0.55	0.57
Model Stacking with NN	**75.90%**	**0.73**	**0.71**	**0.75**

The main criteria for choosing the best model are the recall on the "Malignant" class and the F2-score. According to these metrics, the model stacking that uses a Neural Network as meta-classifier is the best one and for that reason it is the model used for the final submission.

It is possible to notice that, for both classification task ("Mass"/"Calcification" and "Benign"/"Malignant"), the model stacking approach has improved the performance with respect to the original base classifiers. This was expectable since the meta-classifier is a model that finds the best way to combine the predictions given by the base classifiers, as observed in 6.1.6.

8 REFERENCES

Duggento, A., Aiello, M., Cavaliere, C., Cascella, G., Cascella, D., Conte, G., . . . Toschi, N. (2019). An Ad Hoc Random Initialization Deep Neural Network Architecture for Discriminating Malignant Breast Cancer Lesions in Mammographic Images. *Hindawi*.

Shen, L., Margolies, L., Rothstein, J., Fluder, E., McBride, R., & Sieh, W. (2019). Deep Learning to Improve Breast Cancer Detection on Screening Mammography. *Nature*.

Tang, C.-m., Cui, X.-m., Yu, X., & Yang, F. (2019). Five Classifications of Mammography Images Based on Deep Cooperation Convolutional Neural Network. *ASRJETS*.

Yi, D., Sawyer, R. L., Cohn III, D., Dunnmon, J., Lam, C., Xiao, X., & Rubin, D. (2017). Optimizing and Visualizing Deep Learning for Benign/Malignant Classification in Breast Tumors. *arXiv*.

Yuton, Y., Lamard, M., Conze, P.-H., Quellec, G., Cochener, B., & Coatrieux, G. (2020). Multi-tasking Siamese Networks for Breast Mass Detection Using Dual-View Mammogram Matching. *Springer Nature Switzerland*.