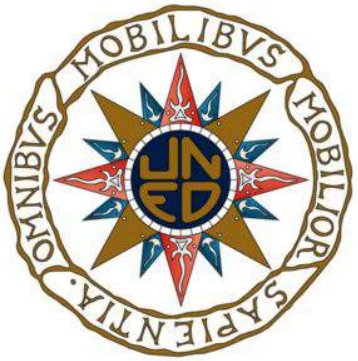
UNIVERSIDAD NACIONAL DE EDUCACIÓN A DISTANCIA (UNED)
ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA INFORMÁTICA



QUALITATIVE ANALYSIS THROUGH VISUAL INTERPRETABILITY TECHNIQUES OF NEURAL NETWORK MODELS FOR MAMMOGRAPHY CLASSIFICATION

Author:
Marta Rodríguez Sampayo

Tutor:
Mariano Rincón Zamorano

MÁSTER UNIVERSITARIO EN INVESTIGACIÓN EN INTELIGENCIA ARTIFICIAL
TRABAJO FIN DE MÁSTER
CONVOCATORIA DE SEPTIEMBRE
CURSO 2020-2021

Contents

List of Tables

QUALITATIVE ANALYSIS THROUGH VISUAL INTERPRETABILITY TECHNIQUES OF NEURAL NETWORK MODELS FOR MAMMOGRAPHY CLASSIFICATION

Marta Rodríguez Sampayo
Departamento de Inteligencia Artificial
UNED
mrodrigue8255@alumno.uned.es

Mariano Rincón Zamorano
Departamento de Inteligencia Artificial
UNED
mrincon@dia.uned.es

ABSTRACT

Nowadays, research in the field of artificial intelligence is focusing on the explainability of the developed algorithms, mainly neural networks. This trend is known as XAI and brings certain advantages such as increased confidence in the decision-making process, improved capacity for error analysis, verification of results and possibility of model refinement, among others. In this work we have focused on interpreting the predictions of recently developed deep learning models through different visualization techniques. The use case we introduce is the detection of breast cancer through the classification of mammographies, since the medical field is widely benefited by the contributions of XAI methods. Furthermore, the target neural networks are based on recent and poorly explored architectures. These are the Vision Transformer model, built through attention blocks, and EfficientNet, designed to improve the performance of convolutional networks.

Keywords Explainable Artificial Intelligence · Interpretability · Deep Learning · EfficientNet · Vision Transformer · Mammography

1 Introduction

The popularity of deep learning continues to rise: new architectures are being designed, novel uses of existing ones are revealed or they are being modified with new computational mechanisms. Dong et al. (2021) surveys several of these advances and describes possible lines of future work. However, for many people, these models remain black boxes, making it difficult to use them in real scenarios beyond experimental assumptions. This situation creates the need to develop techniques for explaining the reasoning that allow the behavior of these systems to be described, so that they can be fully understood even by people with no expertise in this or related fields. The aim is to improve transparency, confidence, fairness and understanding of model bias, objectives pursued by XAI (eXplainable Artificial Intelligence).

According to Barredo Arrieta et al. (2020), a distinction can be made between interpretable (transparent) models and interpretability techniques (post-hoc explanation). In turn, the latter are divided into those model dependent or applicable to several models. Taking into consideration the number of deep learning methods already established and widely used, post-development explanations that are not model-dependent present the opportunity to qualitatively assess the results obtained in different tasks, being able to analyze them from different points of view and, therefore, allowing their improvement beyond the optimization of known metrics such as accuracy or mean square error, for instance.

The use of this type of explanation techniques to improve model development is a considerable advantage that also broadens the perspective on the scope of application, providing complementary information beyond model prediction. The ability to explain AI models in a concise and easy-to-understand manner by non-experts in the field is vital to be able to employ them safely, ethically and reliably. In the field of computer vision, it is common to find studies on the development of methods to determine the regions of the input image used by the model when making predictions, thus making it possible to identify the errors made by the system. Although, their use in real problems is limited to providing

a evidence on whether the model has correctly arrived at the result, similar to the usual accuracy metrics. Thus, their potential is not fully exploited.

On the other hand, medical imaging has become indispensable in the early detection or diagnosis of diseases. Images based on X-rays, tomography, ultrasound and magnetic resonance imaging (MRI) are mainly employed. This is why many researches in artificial vision focus their efforts on the analysis of this type of image datasets, providing different approaches to automatic diagnosis and avoiding human errors committed by the misinterpretation of results due to inexperience or fatigue, among others.However, there are many difficulties in developing diagnostic support systems. Accuracy is particularly important and a rigorous process of validation of results must be carried out, usually by a person who is a specialist in the pathology or imaging method, but not necessarily trained in AI. It is at this point where interpretability methods play a crucial role, allowing to reduce the ambiguity of the learning process and strengthening the credibility of the machine learning models used.

Singh et al. (2020) gather a numerous dataset of different applications of XAI in medical imaging tasks, summarizing the benefits that these approaches bring to both clinical specialists and AI practitioners. One of the cases mentioned in this study is breast cancer. This is one of the most common types of cancer among women whose early detection makes a clear difference in the effectiveness of treatment and, therefore, in mortality reduction. The most widespread type of imaging for preventive check-ups is mammography, although in some cases it is difficult for radiologists to identify lesions and they must refer the patient for a biopsy, a more invasive and costly procedure. Therefore, the development of machine learning models that facilitate the mammography analysis process could be considered almost indispensable. Likewise, allowing the results to be interpreted would facilitate their acceptance by medical professionals. Both Lenis et al. (2020) and Lamy et al. (2019) show clear examples of the benefits that this area of research can bring, allowing to reduce the ambiguity of the learning process and strengthening the credibility of the classifiers used to determine the presence or not of breast cancer in a dataset of mammographies.

The main goal of this work is to perform a qualitative analysis of the predictions of several neural networks based on different learning mechanisms, through visual techniques of interpretability of results. The task to be performed by both models consists in the classification of a dataset of mammographies, differentiating the categories of "malignant" or "benign", including in the latter those images that do not contain signs of malformations or similar findings. The following partial objectives are identified in this process:

- To apply explanation of reasoning techniques to a dataset of deep learning models.
- To analyze the robustness of the system by comparing the results of applying these techniques by making different modifications to the input image.
- To perform a comparison between a convolutional neural network model and a model based on the attention mechanism.
- To train these models on various datasets of different properties, solving classification problems of varying degrees of difficulty.
- To evaluate the potential benefits of adding visual interpretation of results to predictions in breast cancer screening.
- To evaluate the predictions made by both networks from the traditional point of view (quantitative) and based on the results of interpretability techniques (qualitative).
- To document and justify in a reasonable way the choices made when deciding the conditions under which training is carried out in order to allow experimental replicability.

Regarding the structure of this report: firstly, Section 2 presents a brief review of some recent works considered relevant in the field of XAI and Deep Learning, especially those related to the medical field and, particularly, in breast cancer detection. Subsequently, in Section 3, the proposed system and experimental setup are described, where the different datasets of images which have been used are also defined. The experiments conducted, as well as their results and analysis are included in Sections 4 and 5 respectively. Finally, the conclusions drawn are presented in Section 6.

2 Related work

This section has been divided according to the different elements of the system. In Section 2.1 we describe the model interpretability scenario, framing the XAI techniques employed in this work into a particular taxonomy found in the review of work in this area. Whereas a detailed definition of the visual techniques used is provided in Section 2.2. An example image generated with the developed system is presented next to each definition. In each case, an image has been chosen from a dataset to be taken as a basis, and the results and analysis of the mammographies are presented in the corresponding sections (4 and 5). Finally, in Section 2.3 we address the problem of breast cancer detection and diagnosis through deep learning techniques.

2.1 Interpretability

Among the existing possibilities within the field of explainable AI, we have focused on visualization techniques that facilitate the interpretability of model predictions. In other words, it is a matter of explaining in a visual way the relationship between the input images and the result of the classification performed by the network. Colloquially it could be understood as giving an answer to the question *why does this input produce this particular output?*.

This type of post-development explanation has become popular and has led to the creation of a wide variety of methods. There are many taxonomies proposed in the literature consulted to classify them as Gilpin et al. (2018) explain: according to the methodology, the scope of application, the type of information they provide, etc. We can define the four main characteristics of the methods employed as follows:

1. Purpose of the explanation: when making the selection of techniques we have taken into account the role of the end user. Therefore, we have chosen visual techniques that are functional explanations perfectly interpretable by non-AI experts. However they are also valuable for model developers as they help to understand how the network reacts to the input data at different levels.

   The scope of the target also plays a role. In terms of neural networks used for classification, the interest lies in a neuron or output layer associated with the correct class given a sample. That is, local explanations focused on predicting an instance of a class.

2. Explanation control: controllers are understood as the dataset of causal factors whose impact on an aim is described by means of an explanation. The most common are the model input features, which are the ones we use in this research. However, there are XAI methods that include any factor that has some impact on model development, such as training samples, hyperparameter settings or the choice of an optimization algorithm.

3. Explanation families: depending on the way in which the output of the XAI algorithm is generated, these can be grouped into different types or families. The choice of a group of techniques should be made on the basis of their comprehensibility and completeness. Although it will be explained in more detail later, most of the techniques we employ fall into the group of importance scores or relevance heat maps, which combine the attributions of all pixels of the input images.

   Other relevant families are those based on rules or decision trees, dependency graphs, verbal explanations or counterfactuals.

4. Explanation estimation methods: this last categorization of XAI methods takes into account the variation in terms of their applicability to a model and the underlying mechanism. In our case, the techniques used target neural networks, without relying on a specific model, although they have been developed to be used in conjunction with convolutional neural networks, it is possible to use them in other architectures, as discussed in this report.

   The main difference between the chosen dataset of techniques is how the visual explanations are constructed: using the gradient, based on a perturbation mechanism or through activation maps.

In the recently published survey by Zhang et al. (2021), the latest advances in neural network interpretability are concisely summarized, showing the importance of this in terms of model reliability, impartiality of the results and even legality, since in certain research, for example drug design, it is necessary to understand the complete process used.

2.2 Visualization techniques

Tjoa and Guan (2020) describe multiple applications of deep learning models in different fields of medicine that benefit from the use of visualization techniques, such as the detection of brain tumors, classification of Alzheimer's disease pathologies or the classification of melanomas in dermoscopy images. Among the bibliography collected in this compendium, the article by Eitel and Ritter (2019) is of particular interest, where they compare the robustness

of different attribution methods applied to a convolutional neural network trained on a dataset of brain MRI scans of Alzheimer's patients.

The following techniques have been grouped according to the method used to address the problem of assigning a relevance value to each input feature of a model. First, four methods based on gradient computation, all applied to the same example image. Next, a method based on applying a perturbation to the image. And finally, three methods of calculating class activation maps.

2.2.1 Gradient-based techniques

Among the multiple methods for creating visual explanations, those that take the gradient as a base are the simplest to implement, since they depend on the backpropagation mechanism used in the development of neural networks. Their advantages include the speed of execution and their applicability to a wide variety of Deep Learning systems, since they do not depend on the model to which are applied nor on the type of input data involved.

The notation used for the mathematical definitions of each method of this type is as follows: the gradient of the class score function $f(x)$, i.e., the derivative of the output as a function of the inputs, is denoted as $\nabla f(x)$. We will refer to the vector of input features $(x_1, x_2, ..., x_n)$ as x and to a particular class as c.

Vanilla Gradients

It is the simplest visualization method among those used, it is also called saliency map, a concept introduced by Simonyan et al. (2014). It consists of calculating the gradient of the predicted class with respect to the input image. With this technique it is possible to identify the influence of each input pixel on the network output. It is however very sensitive to changes at the pixel level and therefore to input noise.

(a) Original (b) Saliency Map/Vanilla Gradients

Figure 1: Example of vanilla gradients or saliency map.

An example of this technique is shown in Figure 1. The network response is the "dog" label; it can be seen that the lighter areas in Figure 1b correspond to the animal, but there is also noise around it.

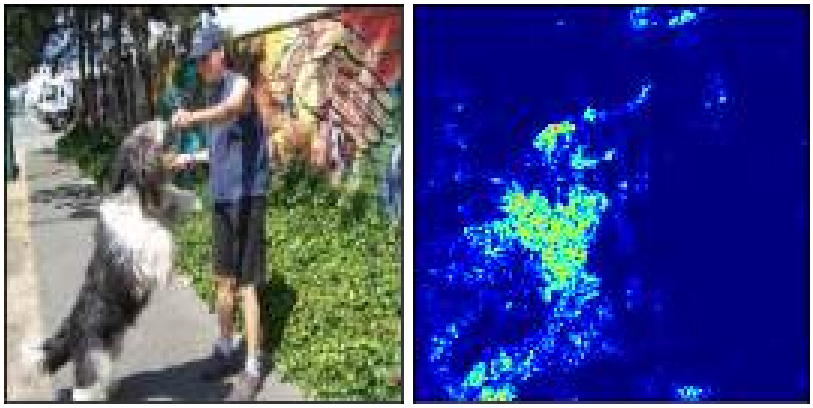
Gradients x Inputs

This attribution technique, defined in Shrikumar et al. (2017), consists of weighting the gradient by the input values. In this way, the importance of each dimension and in what proportion it is shown in the input image is represented. In other words, the effect of any change in the input is reflected through the gradient in the output.

To obtain the individual relevance values, we multiply each gradient by its associated feature, scaling the obtained attribution according to the size of the input:

$$x_i \nabla f(x_i) \quad \forall i = 1...n$$

This method is simple, fast and easy to use, but its scope of application is limited, being reduced to local changes in the input, i.e., analyzing simple functions that behave similarly globally and locally.



(a) Original (b) Gradients*Inputs

Figure 2: Example of Gradients x Inputs.

Using the same example as in the previous case, Figure 2b shows practically the same map as in the case of vanilla gradients but with greater intensity. In this particular case, the figure of the dog is displayed more clearly, although it also amplifies the noise, highlighting non-relevant areas.

SmoothGrad

Seeking to adapt the previous methods to functions of greater complexity, Smilkov et al. (2017) suggests calculating the gradient multiple times, adding each iteration a certain amount of Gaussian noise to the input and averaging the results, so as to smooth out sharp changes in the functions. The mathematical translation of this technique is to compute N gradients of the input by adding a certain amount of noise determined by a normal distribution of standard deviation $sigma$:

$$x_i \frac{1}{N} \sum^{N} \nabla f(x + \mathcal{N}(0, \sigma^2)_i) \quad \forall i = 1...n$$

The effect achieved is the average of the gradients of several inputs very similar to the one we want to explain, keeping the trend of the gradients in a specific area of the input space.

(a) Original (b) SmoothGrad

Figure 3: Example of SmoothGrad.

Looking at Figure 3b one can see the effect of the mentioned smoothing. The intensity of the values obtained is similar to the previous ones, but the outline of the animal is more defined and the surrounding noise is mitigated, although not completely eliminated.

Integrated Gradients

In Sundararajan et al. (2017) the calculated gradients are integrated as the input varies along a linear path from a certain reference point. The idea behind this concept is based on starting from a point with no information, so that any change only adds new data. Usually $x = 0$ is taken as the reference point, so the formula reduces to the following:

$$x_i \int_{\alpha=0}^{1} \nabla f(\alpha x)_i \quad \forall i = 1...n$$

(a) Original (b) Integrated Gradients

Figure 4: Example of Integrated Gradients.

The effect of noise reduction is evident in Figure 4b, where the shape of the animal is drawn almost perfectly and certain areas not belonging to this class are shown more faintly.

Logically, the value of the gradient depends on the type of function to be explained, so these techniques are very sensitive to functions with little variation or very complex. If the target function tends to change very fast, the gradient is only relevant for a small range of input values. Thus, averaging gradients as in SmoothGrad involves obtaining more global explanations. Integrated Gradients also avoids this problem and, simultaneously, that of functions that change little and globally producing small gradients, since it only considers gradients in the shortest direction towards an input that does not provide information.

2.2.2 Perturbation-based techniques

They are applied with the objective of validating that the model correctly identifies the area of interest in the image during the classification task by altering the original input and monitoring the changes in the model predictions.

Occlusion Sensitivity

Proposed by Zeiler and Fergus (2014), it consists of systematically superimposing a dark square region (usually gray) on different portions of the image, occluding them, while monitoring the model output, visualizing the probability of the correct class as a function of the position of that region. As a drawback it can be highlighted that its computational complexity increases proportionally with the number of features, resulting in a slow performance compared to other visualization methods and being highly sensitive to the number of features removed in each iteration.

(a) Original (b) Occlusion Sensitivity

Figure 5: Example of Occlusion Sensitivity.

The result of applying this technique is shown next to the original image in Figure 5. The task to which the network is subjected is the same as in the previous examples, to identify the dog in the image. The result obtained is less clear than in the preceding visualizations since the obtained heat map is superimposed on the original image. Nonetheless, the warmer colored areas are mainly around the animal. Other light areas of the image indicate that applying occlusion hides elements that potentially degrade the performance of the network.

2.2.3 Class activation map-based techniques

As defined in Zhou et al. (2016), class activation maps (CAMs) indicate the discriminative regions of an image used by a neural network, usually CNN, to identify a category. Some of the methods included in this section also base their result on the information provided by the gradient, however they present several common characteristics that differentiate them significantly from the dataset of techniques defined in Section 2.2.1.

In general, these methods are based on obtaining a linear combination of the feature (or activation) maps $A^k_{i,j}$ for each spatial location (i, j) and the weights or attributes w^c_k computed for a certain class c of interest. We denote S^c as the final classification score obtained and $L^c_{i,j}$ as the CAM obtained as a result of the computation. In addition k refers to the k-th value of the feature vector, with a total size of N, i.e. $\forall k = 1...N$.

Therefore, the final calculation in all these techniques is the sum of the activations weighted by the weights obtained, applying the $ReLU$ function to obtain the characteristics with a positive influence on the class of interest.

$$L^c_{i,j} = ReLU(\sum_k w^c_k A^k_{i,j})$$

To obtain the class activation map, a layer of Global Average Pooling (GAP) is added before using Softmax, modifying the network architecture and achieving:

$$S^c = \sum_k w^c_k \cdot \sum_i \sum_j A^k_{i,j}$$

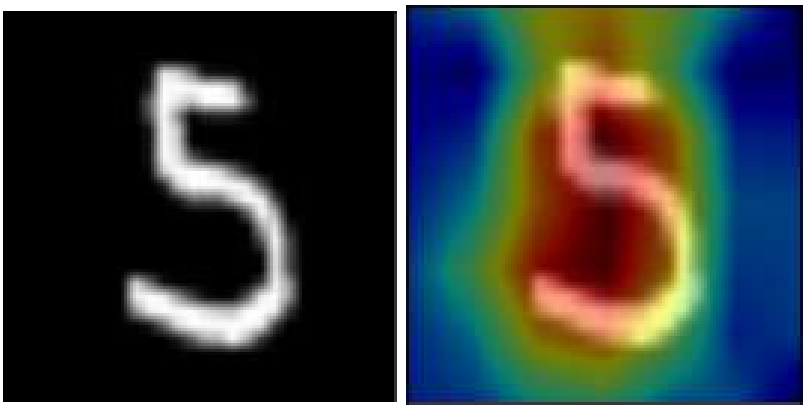
Each of the variants used is defined below.

Grad-CAM

To avoid the need for a GAP layer in the neural network model, as described in the CAM method, Selvaraju et al. (2019) propose to use the class-specific information provided by the gradient to generate the location map of the important regions. Backpropagation is employed to calculate the weights:

$$w^c_k = \frac{1}{N} \sum_i \sum_j \frac{\partial S^c}{\partial A^k_{i,j}}$$

where N is the number of pixels of the activation map.



(a) Original (b) GradCAM

Figure 6: Example of GradCAM.

To provide an example of this technique, a multi-class classification scenario has been used, since Grad-CAM tends to perform worse when the differentiation between classes is not straightforward. This problem will be explained in more detail later. Hence, in Figure 6a a handwritten digit is observed, classified by the network with the label "5". The heat map defined by Grad-CAM completely surrounds the digit, centering the strongest red color on the vertical prior to the characteristic curve of the number five.

Grad-CAM++

Chattopadhay et al. (2018) improve on the previous technique by generalizing the function used and avoiding the tendency to downplay the importance of small region feature maps that tend to obtain small values. In this case, the calculation of the weights is refined, employing $\alpha_{i,j}^{k,c}$ instead of a constant N.

$$w_k^c = \sum_i \sum_j \alpha_{i,j}^{kc} \cdot ReLU(\frac{\partial S^c}{\partial A_{i,j}^k})$$

$$\alpha_{i,j}^{kc} = \frac{\frac{\partial S^c}{(\partial A_{i,j}^k)^2}}{2\frac{\partial S^c}{(\partial A_{i,j}^k)^2} + \sum_a \sum_b A_{a,b}^k \{\frac{\partial S^c}{(\partial A_{i,j}^k)^3}\}}$$

(a) Original (b) GradCAM++

Figure 7: Example of GradCAM++.

Returning to the initial classification example in Figure 7b, this approach indicates that the focus of the model is again around both dogs, focusing on the nose of one and ignoring the person holding them. There is still some noise, as

the heat map includes some warm areas of the background furniture or screen, but it is certainly representative of the decision making process followed.

Score-CAM

In Wang et al. (2020) the instability of using gradient propagation for the calculation of attributions is considered, since the gradient changes abruptly with small variations of the input image even if there are no variations in the resulting prediction. To overcome these drawbacks, the score weights obtained for a specific class through the last layer are used instead, eliminating the noise introduced by the gradient. Once the activations of the last layer of the network are obtained, they are resampled to the original dimensions of the image and their pixel values are normalized to the interval [0,1].

$$A_{i,j}^k = \frac{A_{i,j}^k}{max\ A^k - min\ A^k}$$

Subsequently, each activation map is multiplied by the original image, obtaining an image with a superimposed mask.

$$M^k = A^k \cdot I$$

Once these M images are obtained, they are passed to the network using the Softmax function as output.

$$S_k = Softmax(F(M^k))$$

In this way we obtain the scores for each class, from which we extract the value of the target category representing the importance of the k-th activation map.

$$w_k^c = S_k^c$$

(a) Original (b) ScoreCAM

Figure 8: Example of ScoreCAM.

An example of the heat map generated by the last of the techniques analyzed is shown in Figure 8b. The same image as in the case of *Occlusion Sensitivity* has been used to highlight the accuracy of this method which pinpoints the exact position of the dog's chest with the warmest color, although the rest of its body is also highlighted, corroborating the decision of the network.

2.3 Breast cancer detection

In terms of the development of deep learning applications for breast cancer detection, both architectures uniquely designed for the task of mammography analysis and the use of already known architectures (VGG, ResNet, YOLO, etc.) have been investigated. The use of transfer learning with pretrained models in datasets such as ImageNet and subsequently fitted with mammography images is beneficial in most systems. Other work also includes the use of hybrid models, where CNNs or R-CNNs are used as feature extractors together with a machine learning classifier. Most of the reviewed papers use public image datasets. Although most of them use a single dataset, in some cases several

datasets are used to increase the amount of information and to check the correct generalization of the developed model. The most relevant works done in recent years in this field are summarized in Zou et al. (2019) and Oyelade and Ezugwu (2020).

It is more challenging to find works that include some approach to explaining the underlying logic. Although Grad-CAM is one of the most widely used interpretability techniques. For example to compare the models designed by Pérez et al. (2020) with MobileNetV2, presenting the activation maps by class to support the conclusions about the effectiveness of the proposed networks. In the same way as in Liu et al. (2020b) and Habib et al. (2020), using in the latter both mammography and ultrasound imaging. These approaches employ Grad-CAM to provide a complementary explanation to the reasoning of the models, increasing the acceptance of the models by end users and providing a qualitative analysis of the results. It should be noted that these investigations were published last year, highlighting the novelty of this type of technique and the need for further research in this direction.

Probably due to its recent publication, we have found only one paper using the EfficientNet-B5 architecture to analyze mammographies, Suh et al. (2020). Again in conjunction with the Grad-CAM method to visualize the region of interest recognized by the AI models. Obtaining a mean AUC of 0.954 ± 0.020 in classifying into two categories based on the presence or absence of malignant lesions in the images.

Although some of the previous articles on deep learning techniques applied to mammography analysis mention some approaches that include attention mechanisms, so far the vision transformer model does not seem to have been applied to this task. Even though its effectiveness has been proven in the medical field, these applications are still in early stages of development. Valanarasu et al. (2021) propose the incorporation of a control mechanism to the attention module and a particular training strategy to improve the performance of this architecture in the ultrasonic and microscopic image segmentation task. On the other hand, Hatamizadeh et al. (2021) also deals with segmentation, but in this case of 3D brain MRIs. In the case of Mehta et al. (2020) the aim is breast cancer detection, albeit using images of breast biopsies with hematoxylin and eosin staining to perform the classification. In Shin et al. (2020) a GAN based on the transformer architecture is designed for medical image synthesis. These works report the same major difficulty: the need for a large dataset to perform model training. In most cases the lack of large image datasets is overcome by using transfer learning or hybrid architectures.

In addition, most of the works using this architecture only show the attention map as a visual interpretability technique. This is the motivation of Chefer et al. (2021) to propose an original visualization method designed specifically for this type of network and to compare it with other existing techniques.

3 Materials and Methods

3.1 Datasets

When designing the use case to demonstrate the usefulness of the chosen XAI techniques, the applicability to the real world was taken into account as a decisive factor, i.e. the problem to be solved had to belong to a field in which it is relevant to obtain the benefits that interpretability tools can bring. To this end, we decided to use two datasets of mammographies and to focus on the task of classifying relevant findings into the categories 'malignant' and 'benign'.

Name	Source (Year and country)	Cases	Images	Image size	Image type	Annotations
INbreast	Portugal, 2012	115	410	3328x4084 2560x3328	Digital Mammography (CC y MLO), DICOM standard	Contours and type of abnormality Format: XML
DDSM	EEUU, 1999	2620	10480	Variable	Mammography (CC y MLO)	BI_RADS standard, Pacient age, anomaly boundaries (low accuracy), breast density
CBIS-DDSM	DDSM	1566	10239	Variable	Improvements: DICOM standard and more accurate ROI segmentation	BI-RADS standard, calcification type, distribution, breast density, ROI, etc. Format: CSV
MIAS	United Kingdom, 1994	161	322	1024x1024	Mammography (MLO)	Anomaly type, malignancy, type of tissue, center and circle around the mass

Table 1: Mammography image datasets.

The main characteristics of the most commonly used datasets for this type of task are summarized in table 1. The quality of the images in the MIAS dataset is lower due to their antiquity, both in terms of creation and digitization. For

the same reason they are not annotated with the BI-RADS standard or stored in DICOM format, factors that favor the subsequent processing of the data. In addition, most of the related work that has been reviewed uses either INbreast because of the quality of the images or CBIS-DDSM because of its size and the fact that it has been corrected and adapted to current standards. For these reasons we have selected these last two datasets to perform the experiments. Their main properties are briefly presented below.

3.1.1 CBIS-DDSM (*Curated Breast Imaging Subset of DDSM*)

This dataset is a subset of the DDSM (Digital Database for Screening Mammography), enhanced by Lee et al. (2017) to standardize the data, correct annotations and bounding boxes, and generally format it similar to modern datasets used in computer vision tasks. It contains 753 cases of calcifications and 891 cases of masses, categorized according to the BI-RADS standard[1] It should be emphasized that in the article itself they define divisions into subsets for testing and training, thus allowing comparison of relevant investigations. This division is made on a case-by-case basis, using 20% for testing and the remainder for training.

3.1.2 INbreast

The dataset of images that make up this dataset are fully digital, unlike in the case of DDSM, which are mammographies subsequently digitized through different methods. Therefore, the INbreast images have a much higher quality, although their size is reduced to a total of 115 cases. Its creators Moreira et al. (2011) do not define a division between subsets, but they do make some suggestions, so we randomly choose 90 patients for training and 25 for testing.

Usually the information related to medical images is stored using the DICOM format, which allows the image and related information to be included in the same file. In our case we only use the images as input to the network, ignoring the rest of the information present in the file, so we convert the files to png format. In addition, since a comparison is made between both datasets, we establish a standard size of 224x224, resizing the images of both INbreast and CBIS-DDSM.

3.1.3 Baseline datasets

Since the objective of our work is the use of explanation of reasoning techniques, we have reduced the problem to the most basic task, the binary classification of the images as presented in the dataset, without using the segmentation or other data that may be present. However, we believe it is relevant to use other datasets of images with different features to compare the results of the visualization methods applied to a variety of problems.

MNIST[2] Dataset of images of handwritten digits. It has been chosen since it shares with mammographies the property of being composed of grayscale images. This dataset is widely known and used as a reference for testing and comparing several neural network models. The subsets are given in 600000 samples for training and 10000 for testing. We use MNIST as an example of a classification task with multiple categories.

Cats & Dogs[3] This dataset of images available in Kaggle has been chosen, since it consists of 25000 color images and is intended for binary classification. We have randomly divided it into 25% for testing and 75% for training.

To summarize, we consider the following use cases:

- Binary classification of a small dataset of digital mammograms.
- Binary classification of a medium-size dataset of digitized mammographies.
- Multiclass classification of a large size dataset of grayscale images.
- Binary classification of a large size dataset of color images.

3.2 System architecture

The key idea of this research is to provide a deep learning system that brings more confidence to the user and allows a better understanding of the results to the developer. It also seeks to add value to existing and novel methods without compromising their performance and facilitating a way of coupling with other workflows, without having to make complex modifications. For this purpose, we propose the structure represented in Figure 9. The first flow, depicted in

[1] As specified by Aibar et al. (2011) this is a method of classifying mammography findings that is now considered the universal language in the diagnosis of breast pathology.

[2] http://yann.lecun.com/exdb/mnist/

[3] https://www.kaggle.com/shaunthesheep/microsoft-catsvsdogs-dataset

green, is the usual one in the development of neural networks through supervised learning; a dataset of images is used to train the model, which will then be responsible for making predictions about new images. The second flow, in purple, represents the explanation of the predictions. This explanation is achieved by applying a series of scoring techniques to the model. A value is assigned to each area of the input image based on how much it affects the output and these scores are used to draw a saliency map.

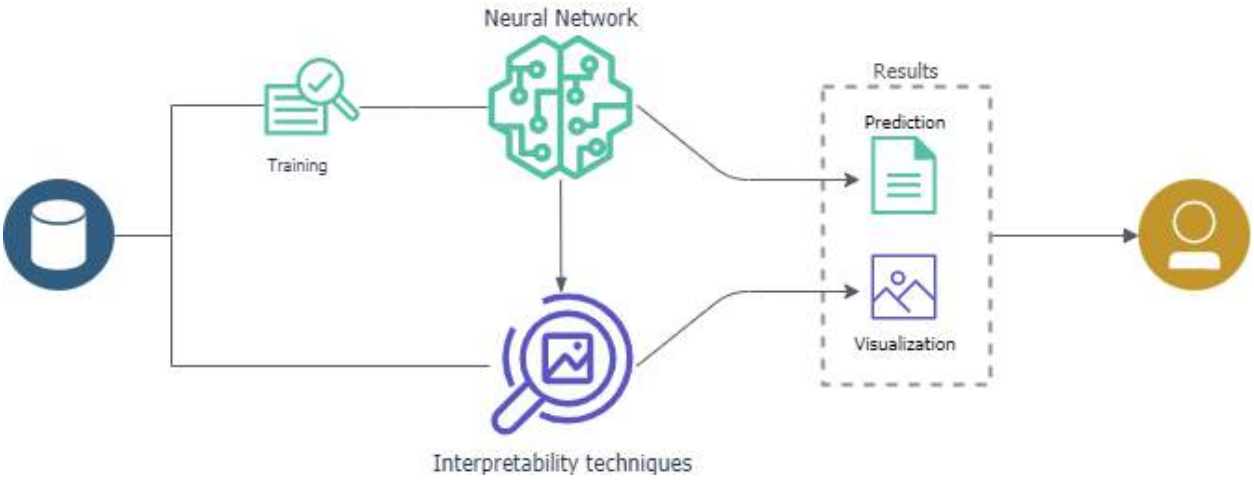


Figure 9: System architecture.

In this way, the end user of the system is presented with two types of responses: the class assigned to the input image and a visual explanation of which areas of this image have influenced the result. This second visualization allows the human being to verify in a simple way whether the network correctly identifies the areas of interest. In the case of breast cancer detection, the specialist would simply have to check for abnormalities in the indicated region to be sure of the correctness of the diagnosis. This explanation provides transparency and confidence, allowing more professionals to apply this type of technique, thus increasing their capacity for analysis. A greater number of simultaneous mammographies can be reviewed in parallel, achieving a preliminary diagnosis, avoiding fatigue and helping to obtain results of equal or greater accuracy in the same amount of time.

Automatic diagnosis itself has the advantage of increasing the accuracy of detecting anomalies in radiographic images. This, together with the explanation provided, would allow previously unnoticed relationships between lesions to be uncovered, and areas highlighted by the interpretability algorithms that would otherwise have gone undetected by the human eye to be repaired.

The basic components that make up this scheme are the recent XAI techniques explained in Section 2.2 and the deep learning architectures used to perform the classification. Similarly to the visualization methods, two of the most recent neural network models have been chosen in order to provide a more complete view of their operation by comparing them. Both models are defined in Section 3.2.1.

3.2.1 Deep learning models

The choice of the two models employed was made taking into account their novelty and the factors that distinguish them from other approaches. On the one hand, the Vision Transformer model stands out because it does not use the convolution operation as is usual when designing a solution for an image classification problem. In contrast, the EfficientNet architecture does consist of convolutional layers, but with an innovative design that makes it a good candidate for this comparison. The origin and main features of both models are detailed below in order to provide functional context.

Vision Transformer

Along with the XAI techniques application, another trend in the field of DL is the incorporation of the attention mechanism in the employed models, especially in the fields of natural language processing and computer vision where its effectiveness has been widely proven (Hafiz et al. (2021); Yang (2020); Niu et al. (2021)). This idea is inspired by

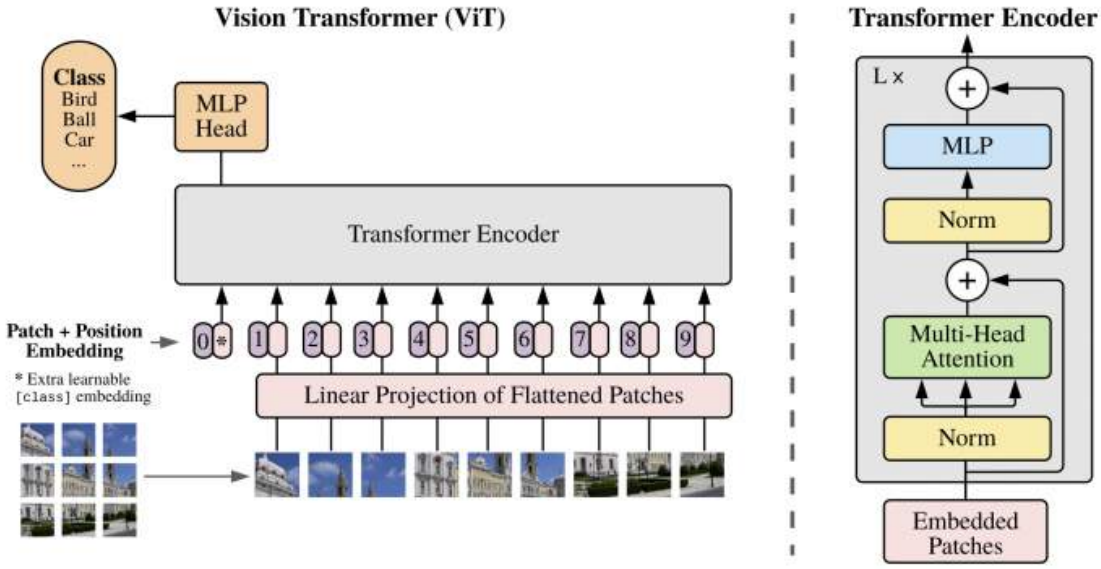


Figure 10: Vision Transformer model structure diagram (Dosovitskiy et al. (2021))

human vision and how we process large amounts of information, concentrating on certain areas selectively and not as a whole.

Initially, this mechanism was applied to existing neural network models, such as convolutional neural networks, helping to improve the results obtained in terms of efficiency and accuracy. Due to the success achieved, in recent years models purely based on the attention mechanism have been developed, such as the Vaswani et al. (2017) Transformer, initially designed for NLP tasks.

Based on the possibilities of this type of architectures, its recent development and the good results reported in different applications, the model defined by Dosovitskiy et al. (2021) called Vision Transformer has been chosen to perform the image classification and to apply the reasoning interpretation techniques mentioned in the previous section.

As shown in Figure 10, the workflow starts by dividing the input images into equal-sized grids or patches, then creating linear representations of reduced dimension of these inputs and feeding them to the encoder, training the model in a supervised way. Its creators propose three different models, varying the size of the patch and the number of blocks that make up the encoder, both the multi-head attention layers and the size of the MLP (Multi-layer perceptron) block, which performs the final classification, and the size of the embedding.

The functional core of this model is the concept of multihead self attention (MSA) which is an extension of the self-attention block commonly used in other neural networks. Broadly speaking, MSA involves executing in parallel k self-attention operations and concatenating their outputs, each of which is called head in the original article. In this way, it is possible to attend to different parts of the sequence in alternative ways in each execution, allowing the model to obtain positional and contextual information.

The main drawback of this type of model is the need for large data datasets to achieve good performance, both in terms of computational cost and accuracy of the results obtained.However, being a recently created architecture, it is to be expected that its design and applications will be profiled as its use increases in different fields as shown in Khan et al. (2021). Thus, performing a qualitative analysis by applying visual interpretation techniques of results on a ViT model can be beneficial in order to better understand its inner workings, beyond the attention maps commonly used for this purpose.

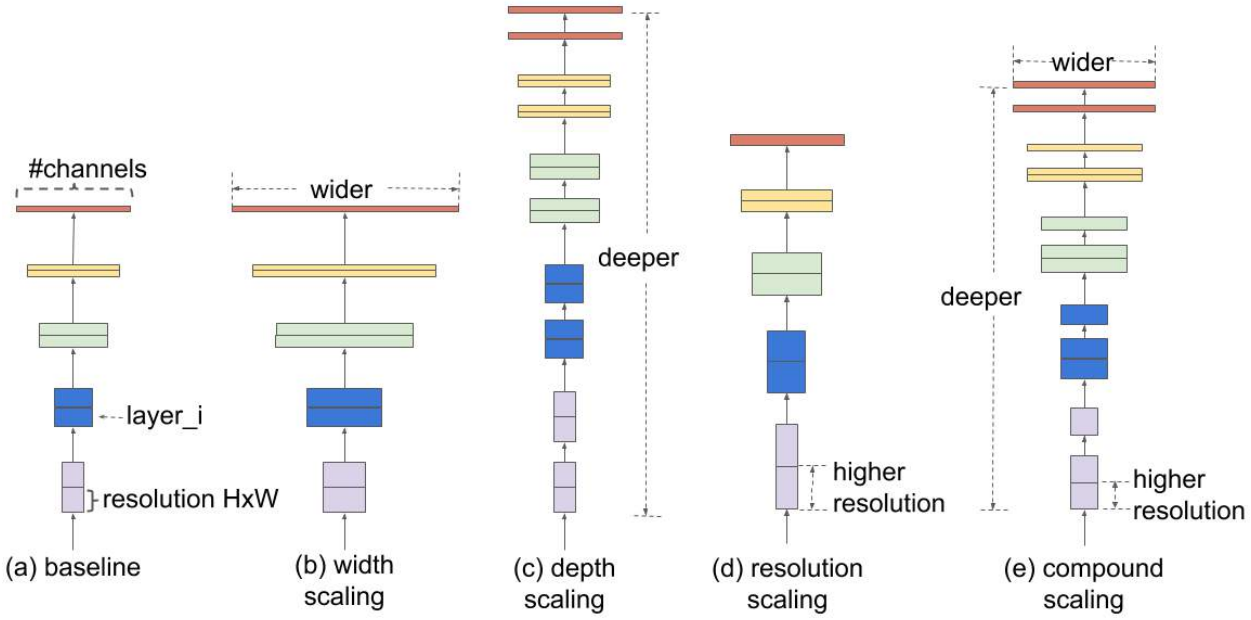


Figure 11: Different neural network scaling methods (Tan and Le (2019))

EfficientNet

In Computer Vision, Convolutional Neural Networks, known by their acronym CNNs, are usually applied, especially to solve classification problems. For this reason, the EfficientNet model has been chosen as a second architecture, since this CNN has been recently designed by Tan and Le (2019). In the same article, the authors claim that the network reaches and, in some cases, surpasses its predecessors in terms of accuracy on benchmark datasets.

The conventional neural network design procedure is based on developing neural models with a fixed resource cost, then arbitrarily expanding the size of the network when more resources become available or increasing the resolution of the input images during the training or evaluation phases, requiring tedious manual adjustment and not always achieving optimal performance. In contrast, the EfficientNet model employs a dynamic scalability method, allowing for improved accuracy without sacrificing efficiency. This technique allows unified scaling of network width, depth and resolution through a fixed dataset of coefficients, the values of which are experimentally determined for each dimension. Figure 11 shows traditional scaling methods (b)-(d) compared to this compound method (e).

The focus in terms of improving convolutional neural networks is not so much on the novelty of the operations performed in the layers that make them up, but rather on increasing their efficiency, trying to reduce the number of parameters and FLOPS (Floating Point Operations Per Second). This translates into a reduction in execution costs, making it possible to use these networks in devices with fewer available resources.

3.3 Experimental setup

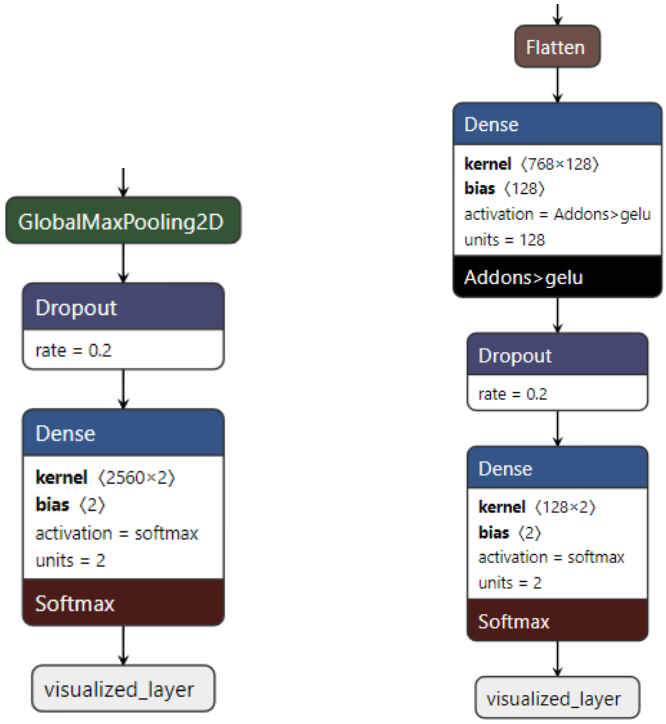
The system on which the experiments have been executed has 4 Tesla V100 GPUs, two with 16GB of RAM and two with 32GB and an Intel(R) Xenon(R) Silver 4210 CPU @ 2.20GHz. In order to facilitate the reproduction of this research, the development has been based on open source software available for its usage, taking into account the indications exposed in `https://blog.ml.cmu.edu/2020/08/31/5-reproducibility/`. The programming language used is `Python` and the API `Keras`, included in the Tensorflow library, commonly used to build deep learning models.

The decisions taken regarding the global conditions (hyperparameter values, network design, regularization methods, etc.) under which the experiments have been performed are based on the results obtained by Steiner et al. (2021). In this work different configurations for the training of several Vision Transformer models are proposed, with small adaptations to our particular case in order to improve the accuracy of both networks.

3.3.1 Implementation

The EfficientNet implementation is the official one, available in the `tensorflow.keras.applications` package, while a Github repository is used for the Vision Transformer based on the official development but using `Tensorflow` instead of `Pytorch`. Within the possible ViT configurations, ViT-B32 has been chosen which consists of 12 layers with a width of 768, 3072 neurons in the MLP block and 12 attention heads, with a total of 86M parameters and a patch size of 32.

Taking into consideration the recommendations of the articles consulted, it has been decided to follow the Transfer Learning strategy, consisting of using pre-trained models, in this case on the Imagenet dataset.The chosen models allow the download and use of the parameters obtained from a previous training on Imagenet. Once loaded, these network layers are frozen and a classifier is added at the end, we have chosen to vary this classifier for each of the cases, both architectures are shown in Figure 12.



(a) Block added to the EfficientNetB5 model

(b) Block added to Vision Transformer model

Figure 12: MLP (MultiLayer Perceptron) blocks used for classification.

Activation functions

The activation functions utilized are:

- *Softmax*: it is used in the last layer of the models since it allows a probability distribution to be obtained as output. This means that the number of neurons in the final layer is equal to the number of classes of the problem and each entry in the vector generated corresponds to the probability that a given input sample belongs to a particular class.
- *GELU (Gaussian Error Linear Unit)*: this activation function was introduced by Hendrycks and Gimpel (2016) with the idea of combining the dropout effect and the popular linear ReLU function. It can be understood as a way of smoothing the effect of the latter by weighting the inputs by their value instead of their sign.

Regularization

Two training regularization strategies, included as layers within the model, are employed:

- `Dropout`: this regularization mechanism, which consists of discarding some neurons randomly during training, thus avoiding overfitting, was proposed by Srivastava et al. (2014). This technique introduces a new hyperparameter that controls the probability of discarding neurons, the value of which is dataset to 0.2 after several experiments.
- `GlobalMaxPooling`: During the pooling operation, downsampling occurs, which consists of replacing the model output with a statistical summary of nearby outputs. In the case of max pooling, the maximum value of a rectangular subregion of the input is selected and, in our case, with global max pooling, the entire input is used to calculate this maximum as output. This operation makes the output representation invariant to small translations of the input, simultaneously reducing its dimensionality. Several possible uses and benefits of this type of layering are discussed in Christlein et al. (2019).

Although the dropout method is included in both blocks, the `GlobalMaxPooling` layer is only part of the layers added to EfficientNet by benefiting from the convolutional layers of EfficientNet and converting the features extracted by these into fixed-size embeddings.

Data augmentation

Usually, in particular in the case of transformers as demonstrated by Steiner et al. (2021), better results, in terms of the calculated metrics, are achieved by using a larger number of images. Since some of the datasets are small in size, we employ data augmentation techniques to provide the network with a larger number of images. The class `ImageDataGenerator` allows to easily make modifications to the original images and introduce them in the final dataset on which the model is trained, among the possible ones we have chosen: random rotation in a range of 20º, height and width shifts in a range of 0.2, cropping following an angle of 0.15 clockwise, random zoom of range 0.15 and random horizontal flips.

Training considerations

Tests have been performed with three different optimizers: on the one hand, SGDW and AdamW proposed by Loshchilov and Hutter (2019) and, on the other hand, RAdam by Liu et al. (2020a), with the parameters described in the previous section. Although the two variants of Adam reached high values of accuracy in the first stages of training, in some cases it was not possible to converge, obtaining a model unable to generalize properly, therefore the results obtained with SGDW, more stable thanks to the use of momentum, are presented.

The main metric used during training is categorical accuracy, while losses are calculated using categorical cross-entropy, as usual. The size of the image batch varies depending on the size of the dataset used being 16, 32 or 64. However, all images are resized to 224x224 in RGB format in order to use the same input tensor in the neural networks and to avoid possible alterations in the results.

Two callbacks of `Keras` are used to monitor the process: on the one side `EarlyStopping` with a patience of 50 epochs and a minimum delta of $1e-4$ on the precision on the validation dataset, this function stops the training when the number of configured epochs elapses without variations in the value of this precision and once finished it loads into the model the weights with which the highest precision was obtained. On the other side, `ModelCheckpoint` periodically saves the weights in a file with h5 extension to be loaded later or to be used in other developments.

The total duration of the training is dataset at 300 epochs, but in most cases, learning stalls early and the function mentioned in the previous paragraph stops the process.

The values of the hyperparameters are decided experimentally, performing repeated trials with variations to improve training performance. The function of each is specified below:

- Weight decay: this parameter decreases the contribution of the coefficients (weights) to the loss function, modifying the function to be optimized and allowing a local optimum to be found. This prevents overfitting.
- Momentum: it is used with the SGD optimizer and, roughly speaking, it accumulates the gradient of previous stages (in a certain percentage), in addition to using the gradient of the current stage, to guide the search by determining the direction to follow. This method achieves faster convergence speed.

- Clipnorm: in this case, the technique used is called Gradient Clipping, specifically Gradient Norm Scaling, and consists of modifying the derivative of the loss function to obtain a given vector norm when the L2 norm of the gradient exceeds a threshold value. This regularizes the behavior of the gradient vector update, decreasing the probability of obtaining disproportionately high or low values.

The values finally used are: weight decay = $1e-4$, momentum = 0.9 and clipnorm = 0.001.

Instead of using a learning rate with a fixed value, a scheduler is used to decrease its value as training progresses. At each step of the optimizer, a cosine decay function with restarts (Loshchilov and Hutter (2017)) is applied to the learning rate. The factor m employed to lower the learning rate decreases from 1 to α (preconfigured minimum value), for s optimization steps, after which a warm restart is applied for t times as many steps and with a m times smaller learning rate. The values used in this function for these parameters are: $m = 1$, $t = 2$, $alpha$ = $1e-6$, initial learning rate = $1e-3$ and $s = 10e4$.

3.3.2 Visualization techniques

Once the network has been trained, the weights with which the best accuracy was obtained on the validation subset during training are loaded and applied together with the interpretability methods on an image extracted from the test subset[4]. In addition to the original image, the `Python` OpenCV library is used to apply certain transformations to it and study the results, these modifications are explained in more detail in Section **??**.

For the implementation of the techniques based on gradients and occlusion sensitivity we use the library tf-explain, while for the activation maps we employ tf-keras-vis. From the latter repository SmoothGrad and Vanilla Gradients are also applied to compare the results, but as they are very similar these are not included in this report. They have been chosen taking into account that both are open-source and are adapted to `Tensorflow`, being able to be applied directly to a model based on `Keras`. Incorporation into the workflow is straightforward, as they use the already trained model together with an input image, generating the corresponding visualizations.

Some of the techniques require certain customized configuration for their proper functioning:

- Occlusion Sensitivity: the size of the patch applied must be specified, in our case this value is 15. Experiments have been carried out with the values 5, 10 and 15, achieving better results with the latter.
- SmoothGrad: the parameters used refer to the number of samples with noise to be generated from the original image and the standard deviation of the normal distribution to generate such noise. Again on an experimental basis, the values 5 and 0.5 are dataset, respectively. In this case, as the number of iterations increased, the efficiency of the system was compromised and a trade-off had to be found between the efficiency and the result obtained.
- Integrated Gradients: the number of times the gradients are applied, called "steps", is passed to this function. The documentation suggests a value between 20 and 300. As for the previous technique, and due to the wide range of recommended values, different values are tested and finally a number of 20 steps is chosen, which is sufficient to obtain explanatory results.

In the case of GradCAM, GradCAM++ and ScoreCAM implementations, it is necessary to make a small modification in the model. Quoting the documentation of `tf-keras-vis`, when applying the activation function softmax in the last layer of the model, it is possible that the generation of the attention maps is obstructed, so it is necessary to replace this function with a linear activation. For this purpose, an instance of `ReplaceToLinear` is used, although it is possible to define one of your own. Subsequently, an instance of `Score` must be created, to which the corresponding indexes of the class of the target image are passed and returns the score given by the model, in this case we use `CategoricalScore`.

[4]Instead of a division into three subsets, only two subsets are used, one of which is used both for validation during training and for testing.

4 Experimental results

This section presents the results obtained by applying the XAI techniques. To complement them, the loss and accuracy plots obtained on the training and test subsets, the confusion matrix and the values of the F1-Score, precision and recall functions are added. The images will be organized according to the dataset of images to which they belong and the model applied. The different visualization methods have been applied both to the original images extracted from the test dataset and to some variations of them obtained performing the following transformations:

- Color transformations:
  - Log transformation: All pixel values are replaced with their logarithmic values, so that dark pixels are expanded (smaller amplitude) while brighter pixels are compressed to a lesser extent (larger amplitude). The result is an image with the details in dark areas more visible.
  - Gaussian noise addition: noise generated by an $N(0, 1)$ distribution.
  - Speckle noise addition: it is generated by multiplying the original image by the values obtained by a standard normal distribution
  - Addition of a black rectangle: a rectangle of random dimensions and location is generated that hides part of the original image.
- Spatial transformations:
  - Warp affine: it consists of a linear deformatio of the image, including a translation, i.e. a transformation that can be expressed in the form of a matrix multiplication followed by a vector addition. Examples of this type of transformations are rotations or scaling operations.
  - 180º rotation of the original image.

In order to organize the information contained in this section, it has been divided into a subsection for each data dataset and each of them, in turn, into two sections, one for each model. In addition to including the images with the results, specific observations to be taken into account for each one will be added; however, both the global analysis of the interpretability techniques and the comparison between the models are developed in Section 5.

4.1 CBIS-DDSM

None of the proposed models achieve good results on this dataset, compared to works such as that of Al-antari et al. (2020), which report an accuracy of 97.50% on a similar diagnostic task. The key difference is that they focus first on detecting the lesion and then perform classification. Nevertheless, such systems have a higher complexity design, applying several preprocessing techniques to the images prior to their input to the network and employing hybrid systems for localization, feature extraction and classification with multiple neural networks.

Considering that the objective of this study is the comparison of different methods of visualization of reasoning, the workflow has been simplified, using the dataset images directly as input to the model.In the case of the Vision Transformer, the image is divided into patches of size 32x32 at the input of the network, being images of dimensions 224x224 a total of 49 patches are obtained. In Figure 13 you can see an example of how this division would be done in an image belonging to the CBIS-DDSM.

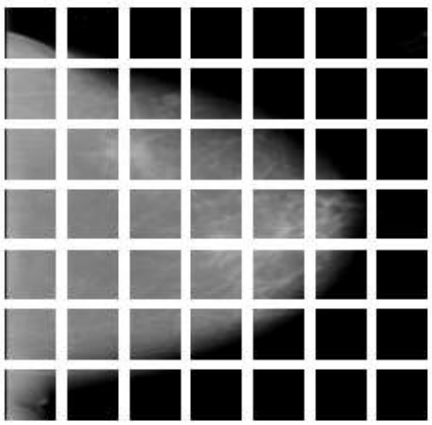


Figure 13: Image divided into network input patches (CBIS-DDSM).

In order to perform a more comprehensive analysis, Figure 14 indicates the area of the mammography where the calcification is located. In an ideal scenario, the network should unambiguously identify this region to correctly assign the corresponding class to the image. Therefore, XAI techniques will indicate whether the model has taken into account the correct area of the image in the classification or not.

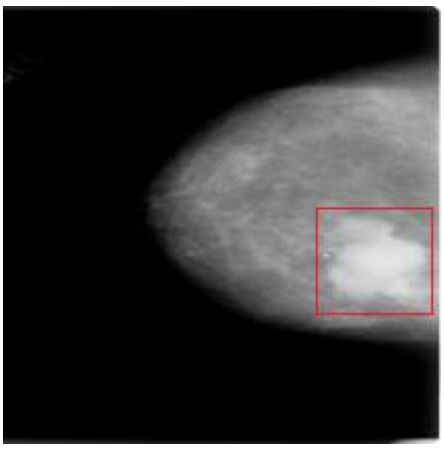


Figure 14: Region of interest segmentation (CBIS-DDSM).

EfficientNet

In the case of this CNN, the accuracy obtained is 69.25%, which is quite low compared to other works mentioned above. The interpretability techniques have been applied to an image of the "malignant" category, i.e. showing signs of a possible breast tumor, such as masses or calcifications. In Figure 16 we can see the results. The first thing that is striking is the results of the last 3 columns representing the CAM-based methods, the most prominent regions are located in the corners of the image, away from the breast itself, where the information needed to perform the classification is located. The occlusion technique shows the most disparate results between the different image variations, while the gradients do not show significant differences in general.

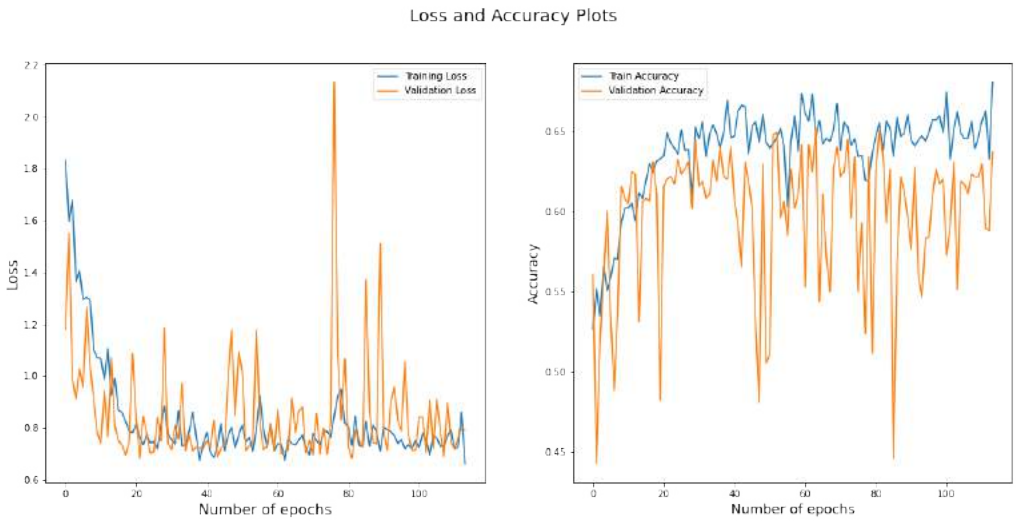


Figure 15: ENet: Loss function and categorical accuracy plots (CBIS-DDSM).

Class	Precision	Recall	F1-Score	Support
Benign	0.68	0.77	0.73	386
Malignant	0.59	0.48	0.53	265
Accuracy			0.65	651
Macro average	0.64	0.63	0.63	651
Weighted average	0.65	0.65	0.65	651

Table 2: ENet: Classification report (CBIS-DDSM)

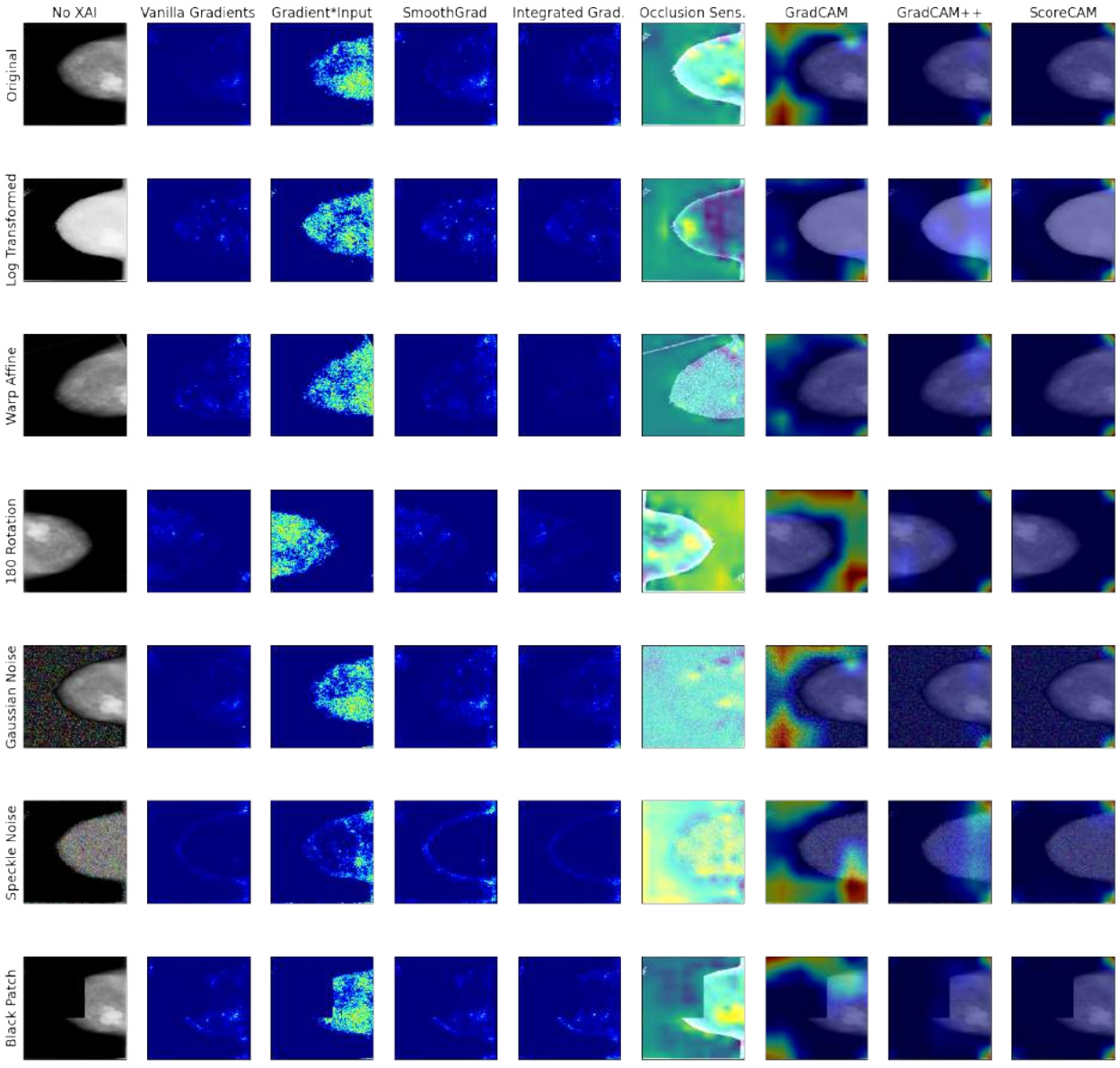


Figure 16: Visualization techniques applied to EfficientNet (CBIS-DDSM).

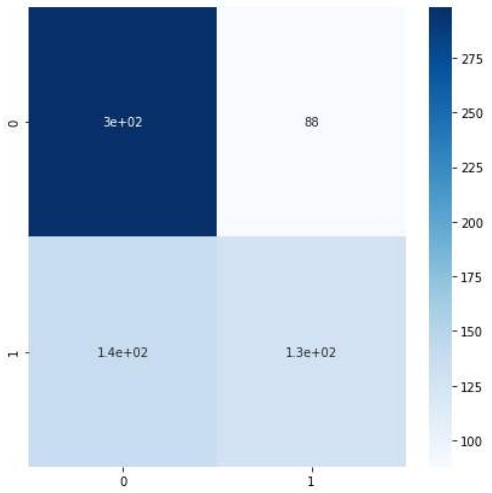


Figure 17: ENet: Confusion matrix (CBIS-DDSM)

Vision Transformer

This model achieves 59.91% classification accuracy, significantly worse than EfficientNet and the reviewed literature, even close to randomness as it is a binary problem. However, techniques based on activation maps show breast-centered regions of interest, although very sensitive to the transformations applied to the target image, as can be noticed in Figure 19. In contrast to the Gradient*Input algorithm, all other gradient-based methods produce positive results in the black areas of the image. The reason why this technique differs from the others is logical, since it multiplies the importance values obtained by the input pixels whose values are higher in the lighter areas. Also note the square areas that form the patches into which the image is divided at the entrance of the network that form small units with their own attribution maps.

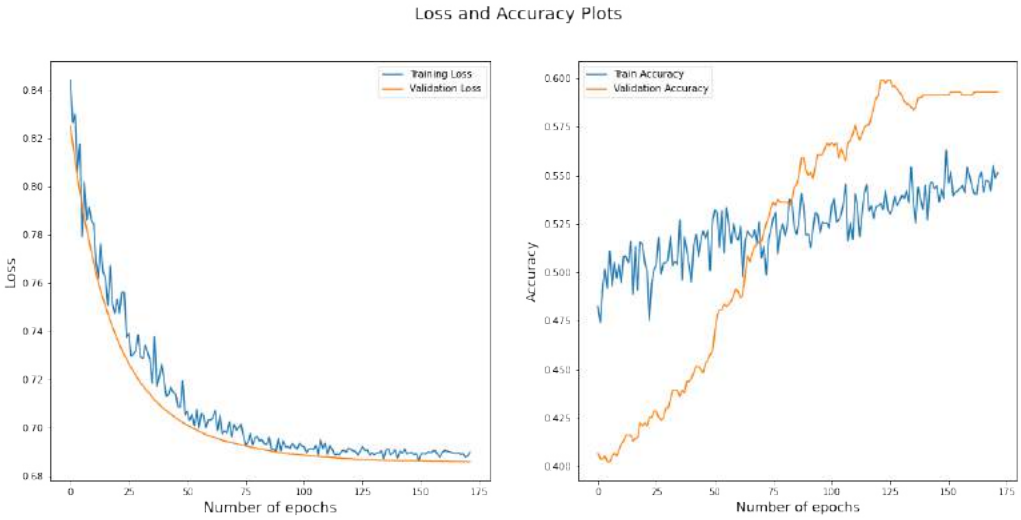


Figure 18: ViT: Loss function and categorical accuracy plots. (CBIS-DDSM)

Class	Precision	Recall	F1-Score	Support
Benign	0.60	0.94	0.73	386
Malignant	0.54	0.11	0.18	265
Accuracy			0.60	651
Macro average	0.57	0.52	0.46	651
Weighted average	0.58	0.60	0.51	651

Table 3: ViT: Classification report (CBIS-DDSM)

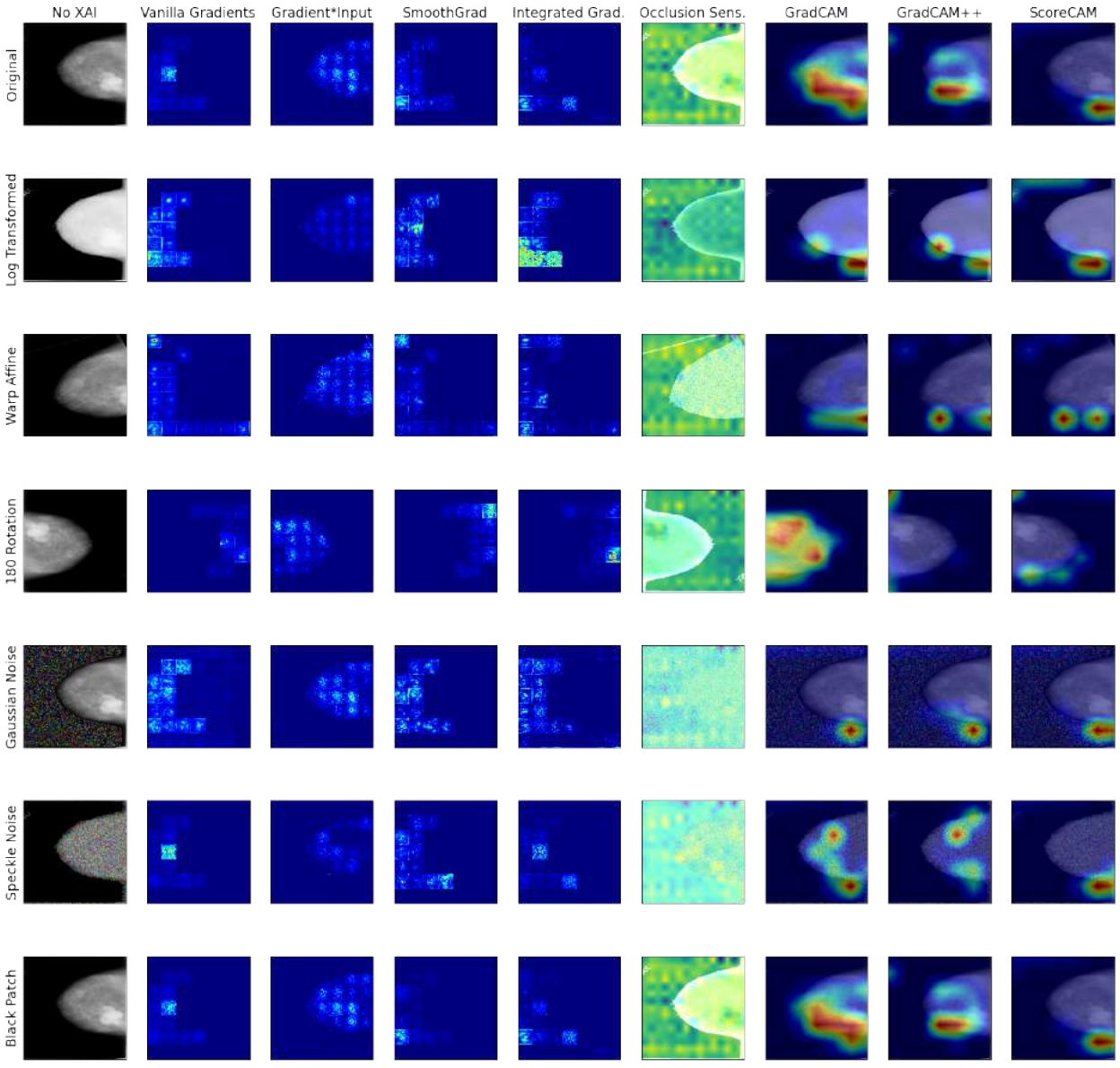


Figure 19: Visualization techniques applied to Vision Transformer (CBIS-DDSM).

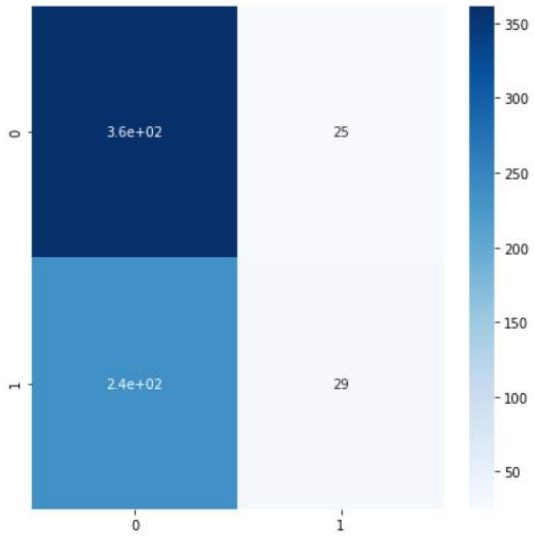


Figure 20: ViT: Confusion matrix (CBIS-DDSM)

4.2 INbreast

Al-antari et al. (2020) use the same system with INbreast obtaining 95.32% on the same binary problem. These images have a much higher resolution as they were created digitally, yet we reduced their size to use exactly the same configuration between datasets. On the other hand, we also did not obtain comparable results with the models used for the same reasons as with the previous dataset. Again, an example of input to the ViT model from this image dataset is shown in Figure 21.

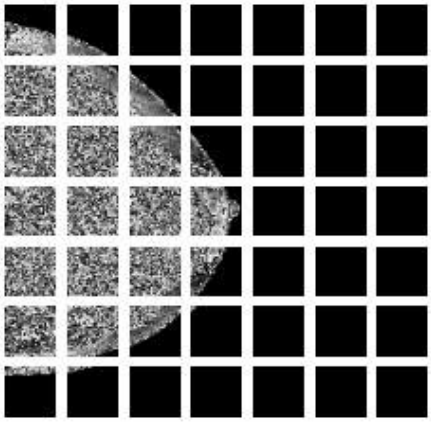


Figure 21: Image divided into network input patches (CBIS-DDSM).

In the chosen CBIS-DDSM image it was easier for the untrained eye to identify the region with the anomaly, in this case it appears less obvious, as reflected in Figure 22. Applying interpretability techniques on this occasion also serves to test whether the models encounter a difficulty analogous to that of the human eye in recognizing these areas of interest.

Figure 22: Region of interest segmentation (CBIS-DDSM).

EfficientNet

The obtained accuracy is 68.18%, slightly higher than the classification result on the CBIS-DDSM. The results of the gradient saliency maps in the natural and integrated gradients stand out, due to the low intensity of the attributions on the image. In this case it seems that the results of the last columns of Figure 24, corresponding to the CAM-based methods are closer to the area where the mass is located than in the case of the previous dataset of images.

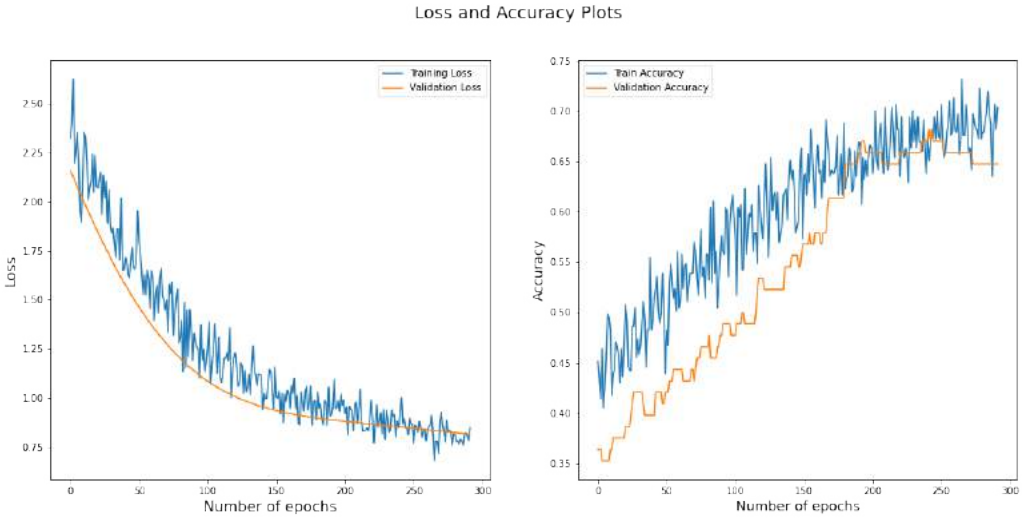


Figure 23: ENet: Loss function and categorical accuracy plots (INbreast)

Class	Precision	Recall	F1-Score	Support
Benign	0.71	0.92	0.80	62
Malignant	0.38	0.12	0.18	26
Accuracy			0.68	88
Macro average	0.54	0.52	0.49	88
Weighted average	0.61	0.68	0.62	88

Table 4: ENet: Classification report (INbreast)

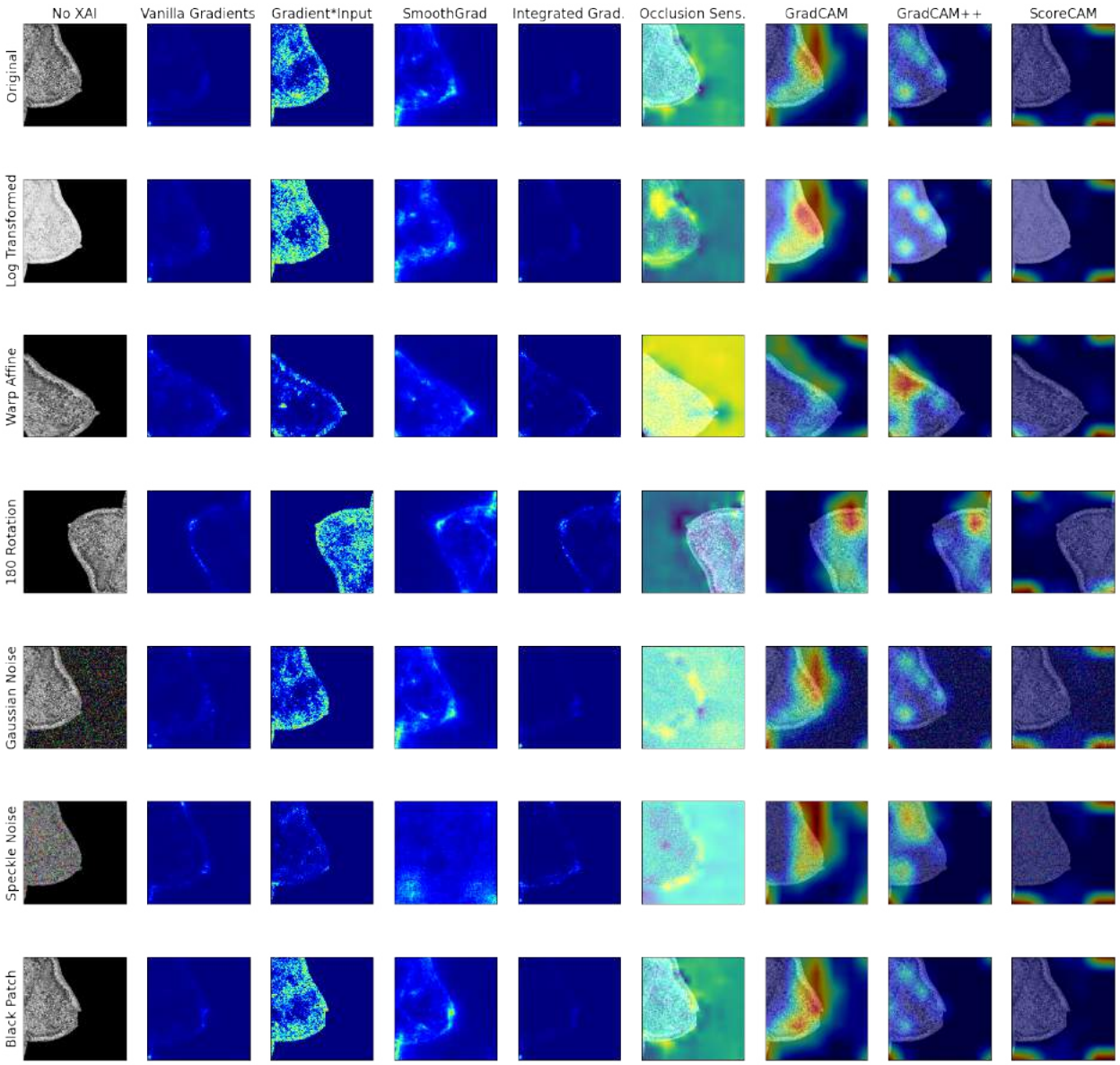


Figure 24: Visualization techniques applied to EfficientNet (INbreast).

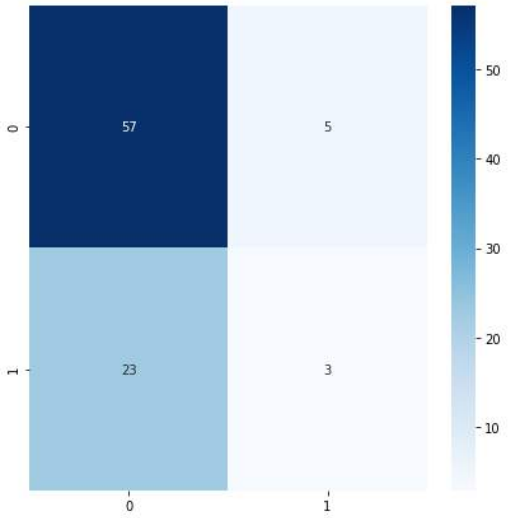


Figure 25: ENet: Confusion matrix (INbreast)

Vision Transformer

In this case the transformer architecture achieves 72%, surpassing the CNN, but remaining quite far from the state of the art. The visualizations are very similar to those obtained with the CBIS-DDSM image dataset, but with higher values of importance translating into warmer colored areas in the CAMs and more intense points in the gradient-based ones.

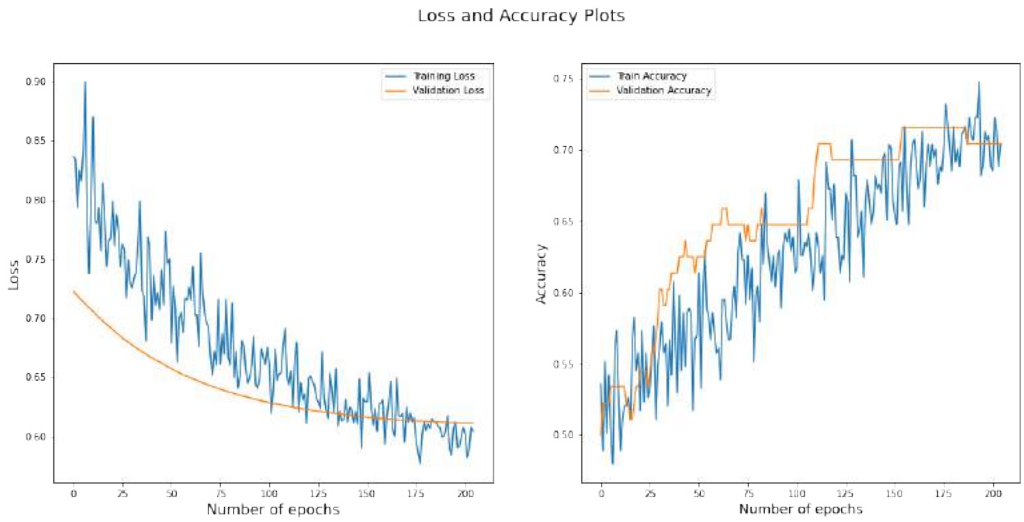


Figure 26: ViT: Loss function and categorical accuracy plots (INbreast)

Class	Precision	Recall	F1-Score	Support
Benign	0.71	1.00	0.83	62
Malignant	1.00	0.04	0.07	26
Accuracy			0.72	88
Macro average	0.86	0.52	0.45	88
Weighted average	0.80	0.72	0.61	88

Table 5: ViT: Classification report (INbreast)

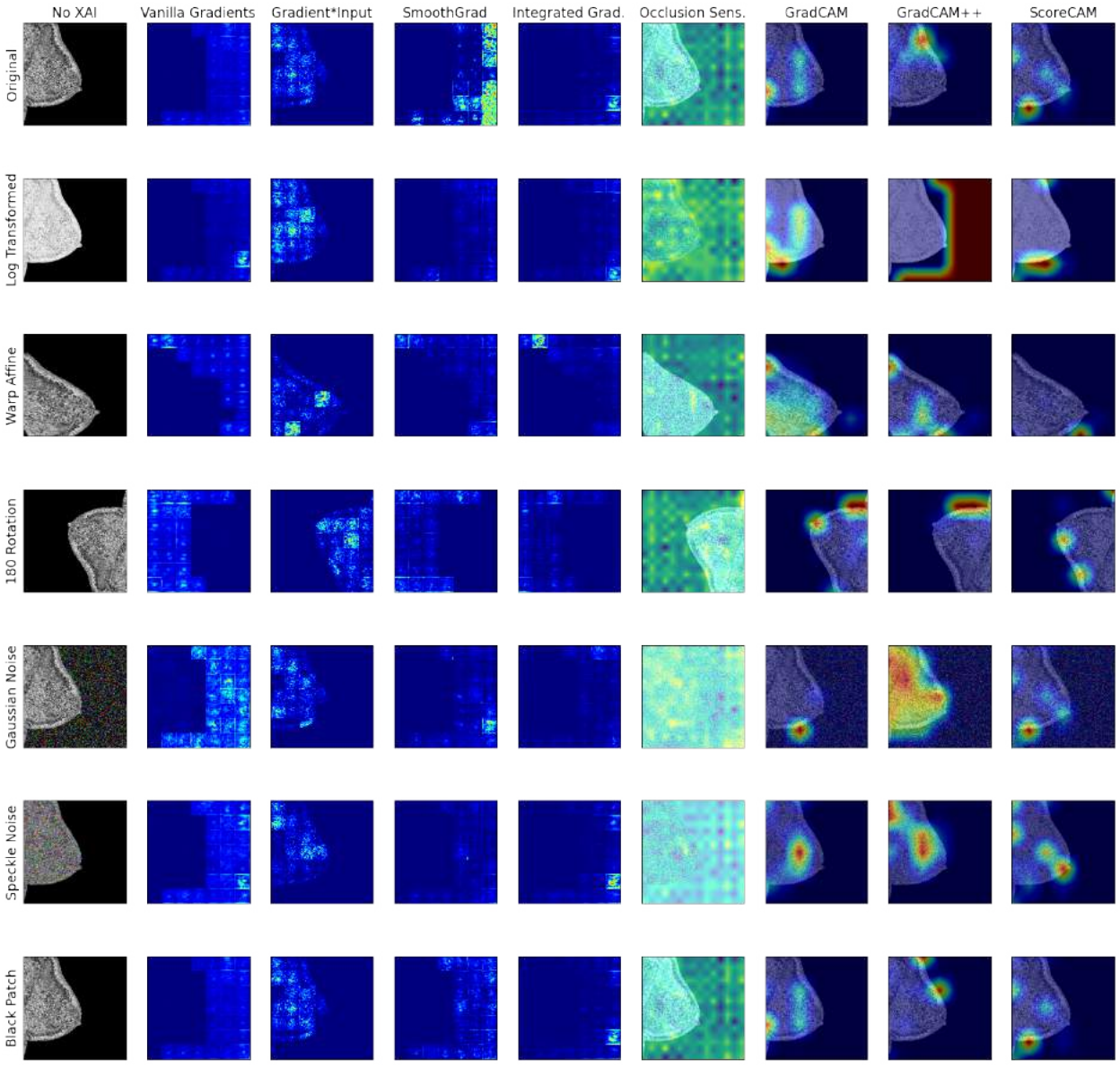


Figure 27: Visualization techniques applied to Vision Transformer (INbreast).

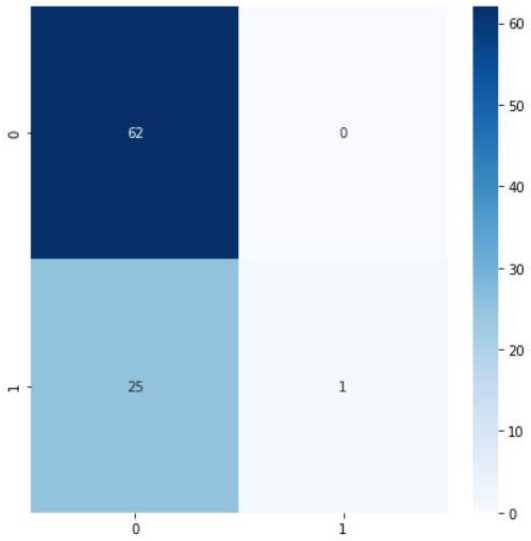


Figure 28: ViT: Confusion matrix (INbreast)

4.3 MNIST

The best classifier published on the MNIST image dataset[5] achieves an accuracy of 99.87%. Taking this value as a reference, the results obtained by the models used are mediocre, but the objective in this case is to check which areas of the input image affect the decision taken in a classification problem with multiple categories through the proposed visualization methods. The images in this dataset are 28x28 in size, smaller than the patch used in the ViT, so they have been increased in size in order to correctly perform the division, as shown in Figure 29..

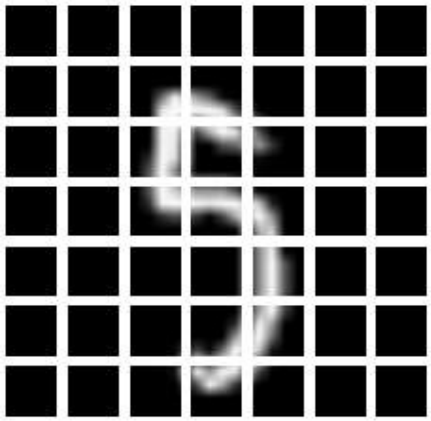


Figure 29: Image divided into network input patches (MNIST).

[5] https://paperswithcode.com/sota/image-classification-on-mnist

EfficientNet

Taking into consideration the previous cases included in this report, it can be considered that the 96.05% accuracy obtained by the ENet model in this dataset implies a good classification performance. As for Figure 31 we can highlight the results of ScoreCAM, in which the represented digit does not seem to be correctly located. In contrast, the result of GradCAM, in addition to focusing on the representation of the number remains constant in several of the different transformations applied.

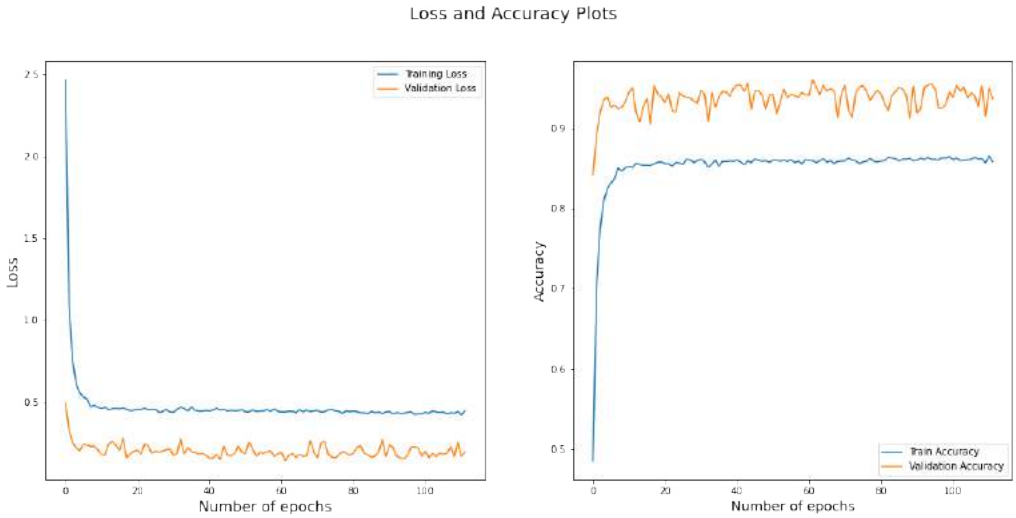


Figure 30: ENet: Loss function and categorical accuracy plots (MNIST)

Class	Precision	Recall	F1-Score	Support
0	0.98	0.99	0.98	980
1	0.99	0.99	0.99	1135
2	0.93	0.94	0.94	1032
3	0.93	0.96	0.95	1010
4	0.96	0.97	0.96	982
5	0.94	0.93	0.93	892
6	0.98	0.97	0.98	958
7	0.97	0.94	0.95	1028
8	0.97	0.97	0.97	974
9	0.95	0.94	0.95	1009
Accuracy			0.96	10000
Macro average	0.96	0.96	0.96	10000
Weighted average	0.96	0.96	0.96	10000

Table 6: ENet: Classification report (MNIST)

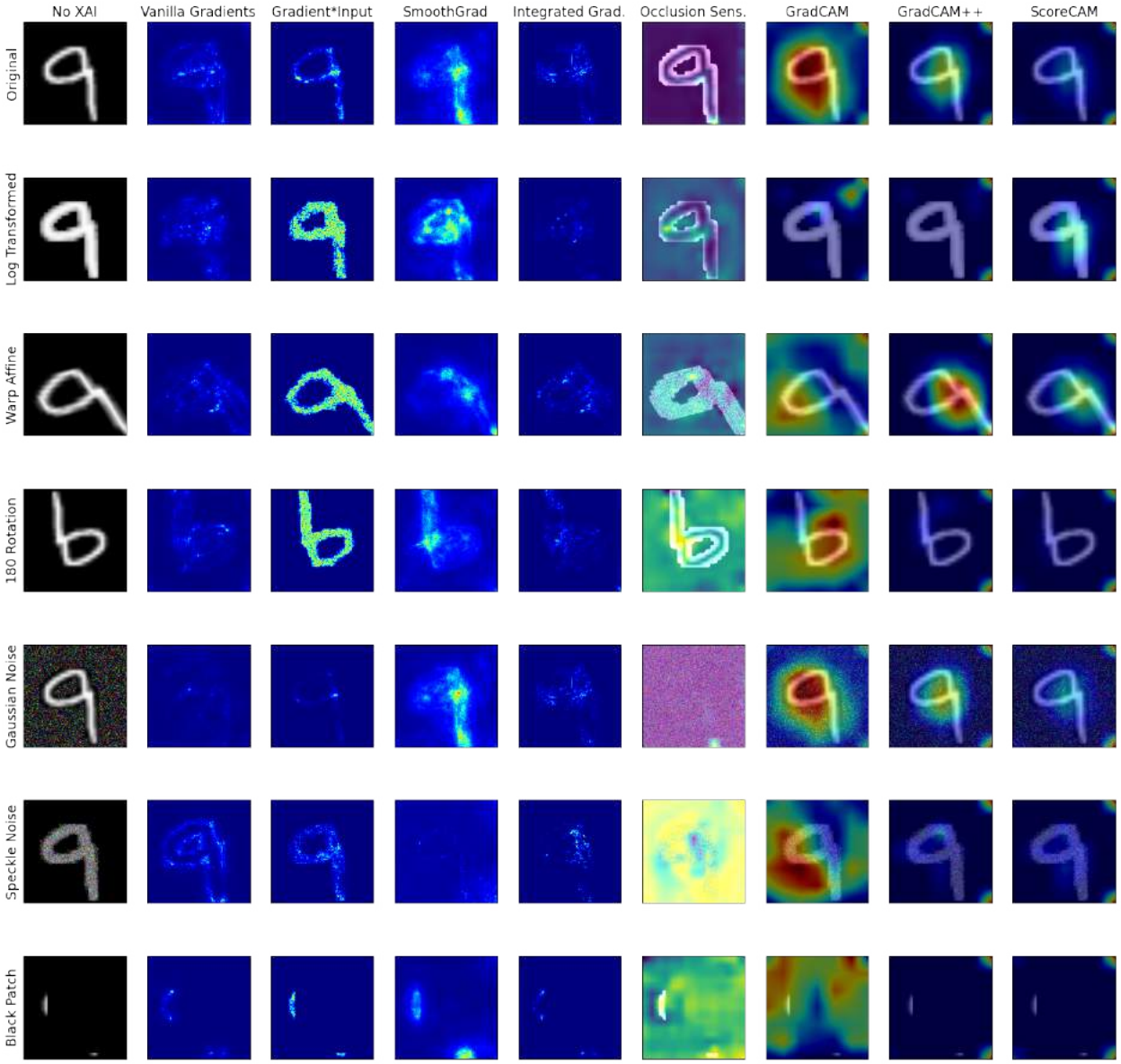


Figure 31: Visualization techniques applied to EfficientNet (MNIST).

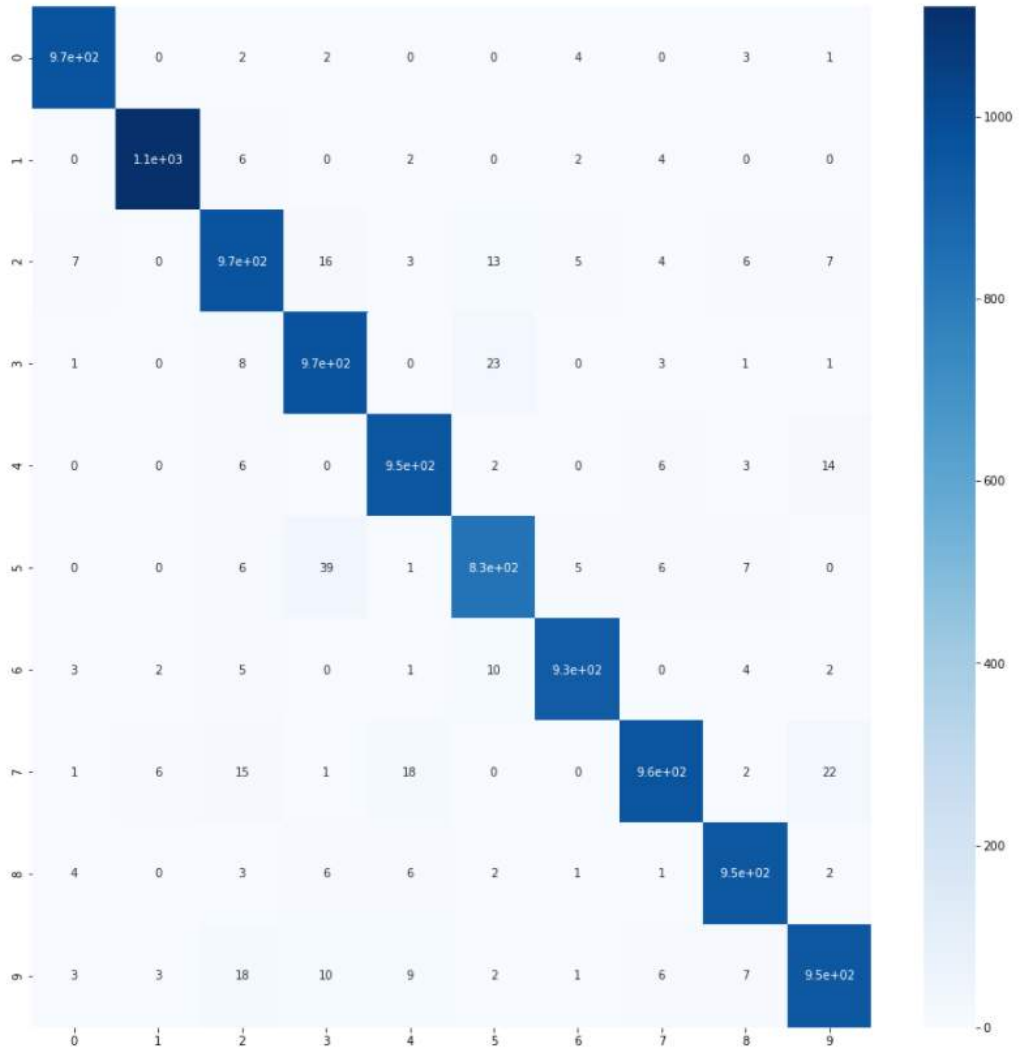


Figure 32: ENet: Confusion matrix (MNIST)

Vision Transformer

Among the experiments performed on the Vision Transformer network, the highest accuracy is obtained using MNIST, with a value of 80.86%. Despite this, there is still a considerable difference between this and the values obtained by other networks on the same task. Looking at Figure 34, the techniques that seem to be most accurate in terms of the calculation performed during prediction are the heat maps produced by the GradCAM, GradCAM++ and ScoreCAM techniques. In addition, image modifications seem to favor the calculations, obtaining maps of higher intensity.

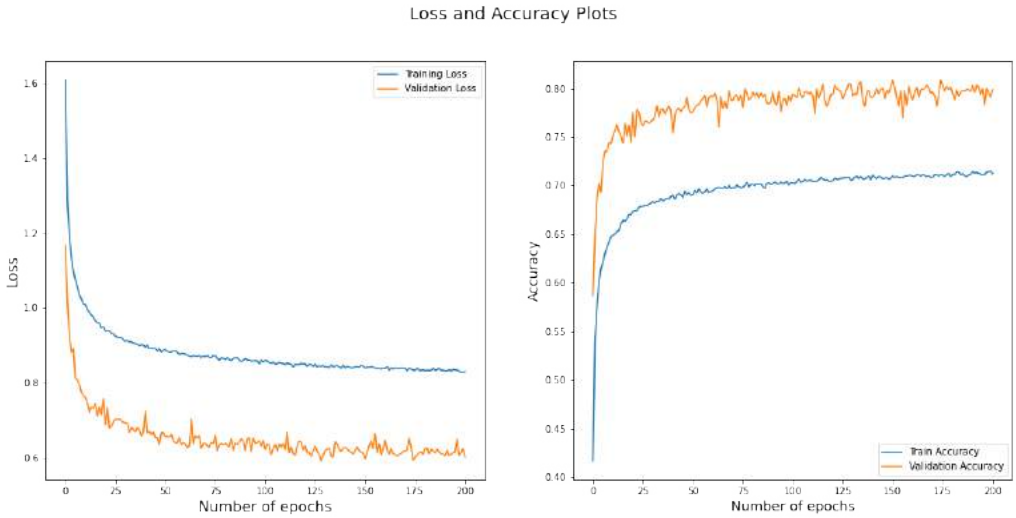


Figure 33: ViT: Loss function and categorical accuracy plots. (MNIST)

Class	Precision	Recall	F1-Score	Support
0	0.81	0.90	0.85	980
1	0.99	0.98	0.99	1135
2	0.68	0.65	0.67	1032
3	0.76	0.79	0.78	1010
4	0.83	0.86	0.85	982
5	0.71	0.66	0.68	892
6	0.76	0.76	0.76	958
7	0.84	0.86	0.85	1028
8	0.85	0.79	0.82	974
9	0.81	0.81	0.81	1009
Accuracy			0.81	10000
Macro average	0.80	0.81	0.80	10000
Weighted average	0.81	0.81	0.81	10000

Table 7: ViT: Classification report (MNIST)

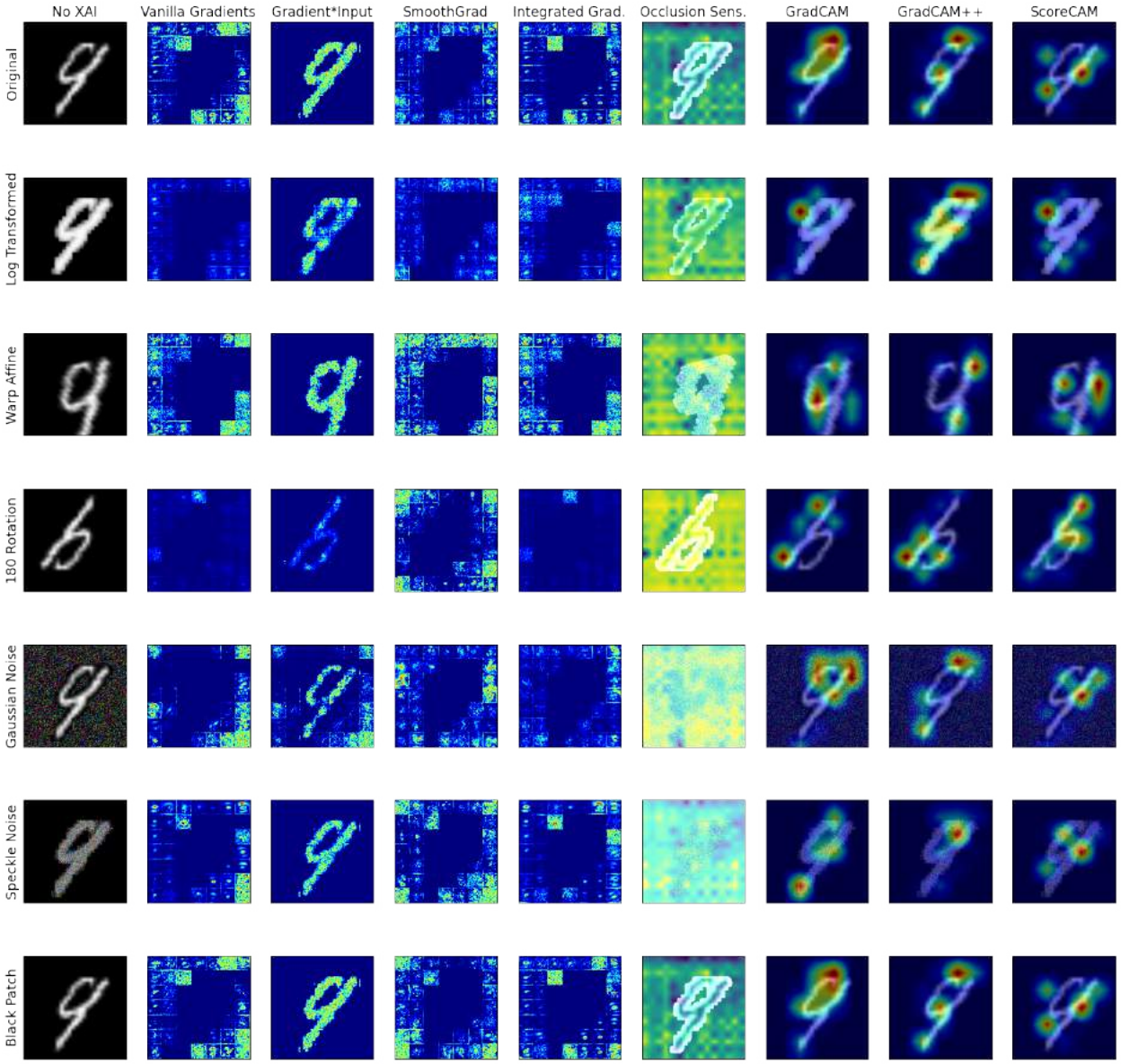


Figure 34: Visualization techniques applied to Vision Transformer (MNIST).

Figure 35: ViT: Confusion matrix (MNIST)

4.4 Cats & Dogs

This case study is the only one with images in RGB format, i.e., with 3 channels in the input image. The task of binary prediction of dog and cat images is common[6] and different accuracy values have been reported over the years, reaching approximately 99% hit rate. Although the images it contains are of different sizes, they have been resized and adapted to the models. In Figure 36 we again observe an example of the segmentation required for training the transformer.

Figure 36: Image divided into network input patches (Cats & Dogs).

[6] A competition in Kaggle was conducted to develop deep learning models that would classify the images in it.

EfficientNet

As might be expected, the EfficientNet-B5 model achieves an accuracy of 97.65%, considerably better than in the mammography datasets and comparable to those obtained by other CNNs in the same task. In this case, Figure 39 shows uniform results for most algorithms, except for the occlusion and Score-CAM techniques that seem to be more affected by the input transformations.

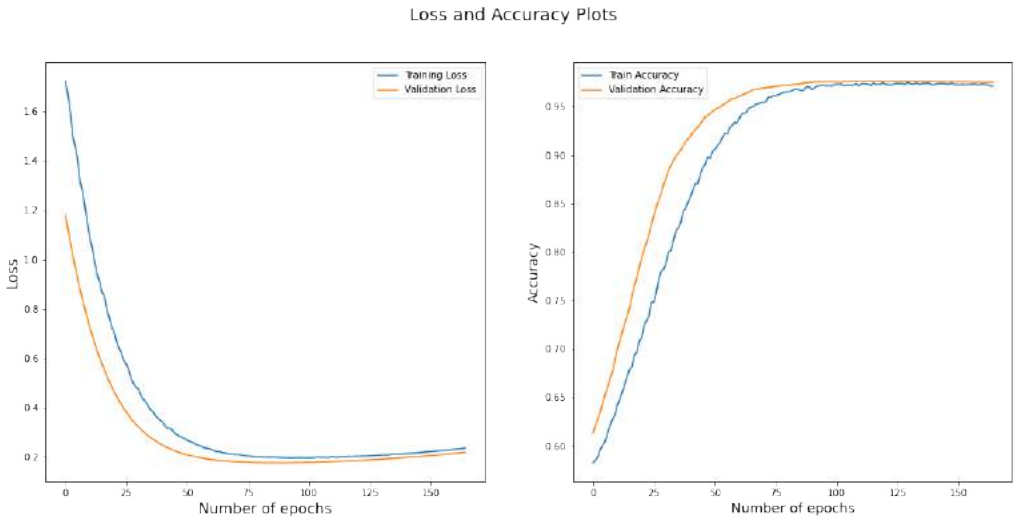


Figure 37: ENet: Loss function and categorical accuracy plots. (Cats & Dogs)

Class	Precision	Recall	F1-Score	Support
Cat	0.99	0.96	0.98	3124
Dog	0.97	0.99	0.98	3124
Accuracy			0.98	6248
Macro average	0.98	0.98	0.98	6248
Weighted average	0.98	0.98	0.98	6248

Table 8: ENet: Classification report (Cats & Dogs)

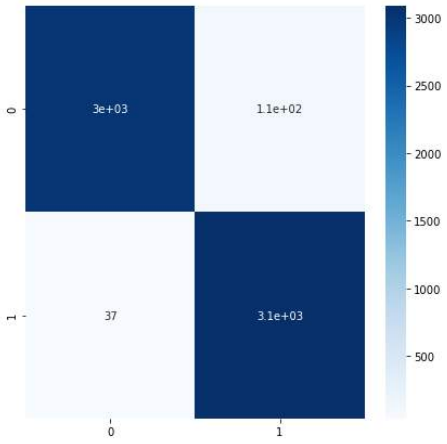


Figure 38: ENet: Confusion matrix (Cats & Dogs)

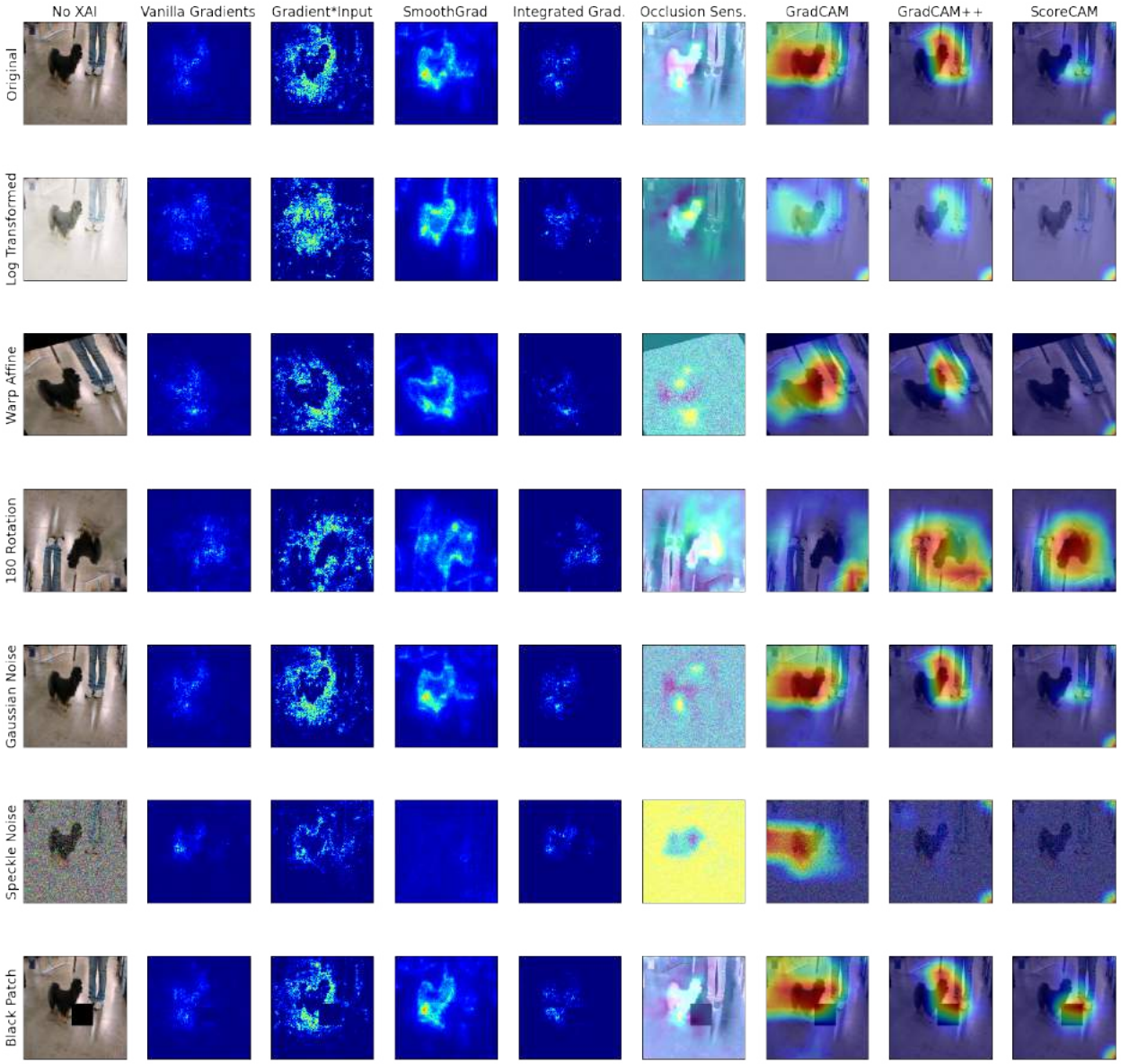


Figure 39: Visualization techniques applied to EfficientNet (Cats & Dogs).

Vision Transformer

Contrary to what might be expected, ViT does not manage to obtain an accuracy close to the models consulted, remaining at 67.22%. The maps obtained by the gradient-based methods are not at all conclusive, showing images with hardly any information in the corresponding columns of Figure 42.

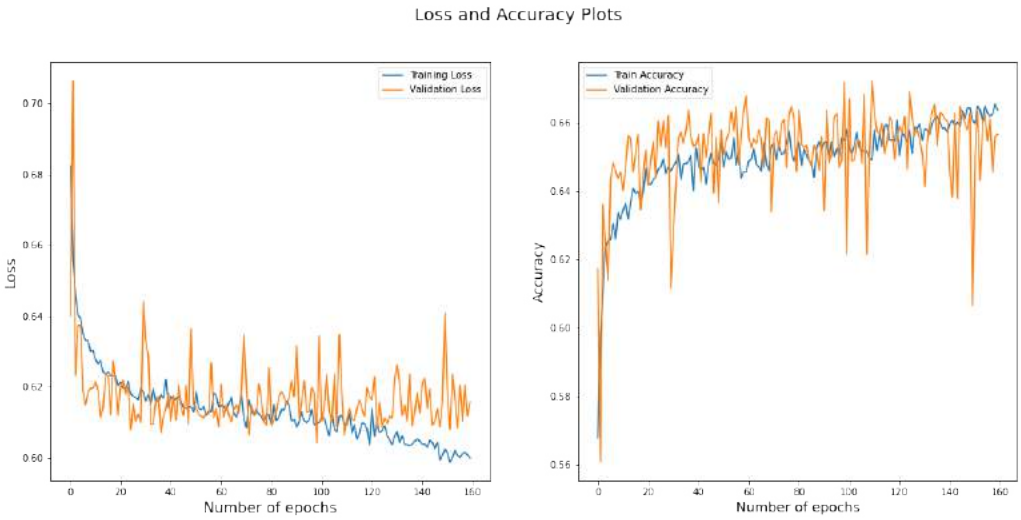


Figure 40: ViT: Loss function and categorical accuracy plots (Cats & Dogs)

Class	Precision	Recall	F1-Score	Support
Cat	0.67	0.68	0.68	3124
Dog	0.68	0.66	0.67	3124
Accuracy			0.67	6248
Macro average	0.67	0.67	0.67	6248
Weighted average	0.67	0.67	0.67	6248

Table 9: ViT: Classification report (Cats & Dogs)

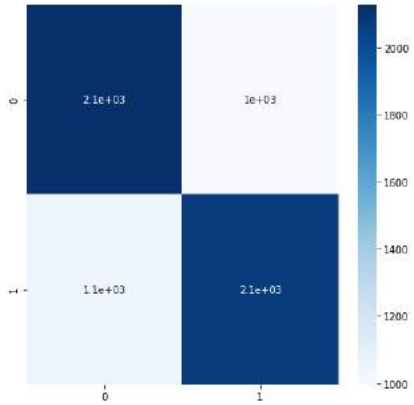


Figure 41: ViT: Confusion matrix (Cats & Dogs)

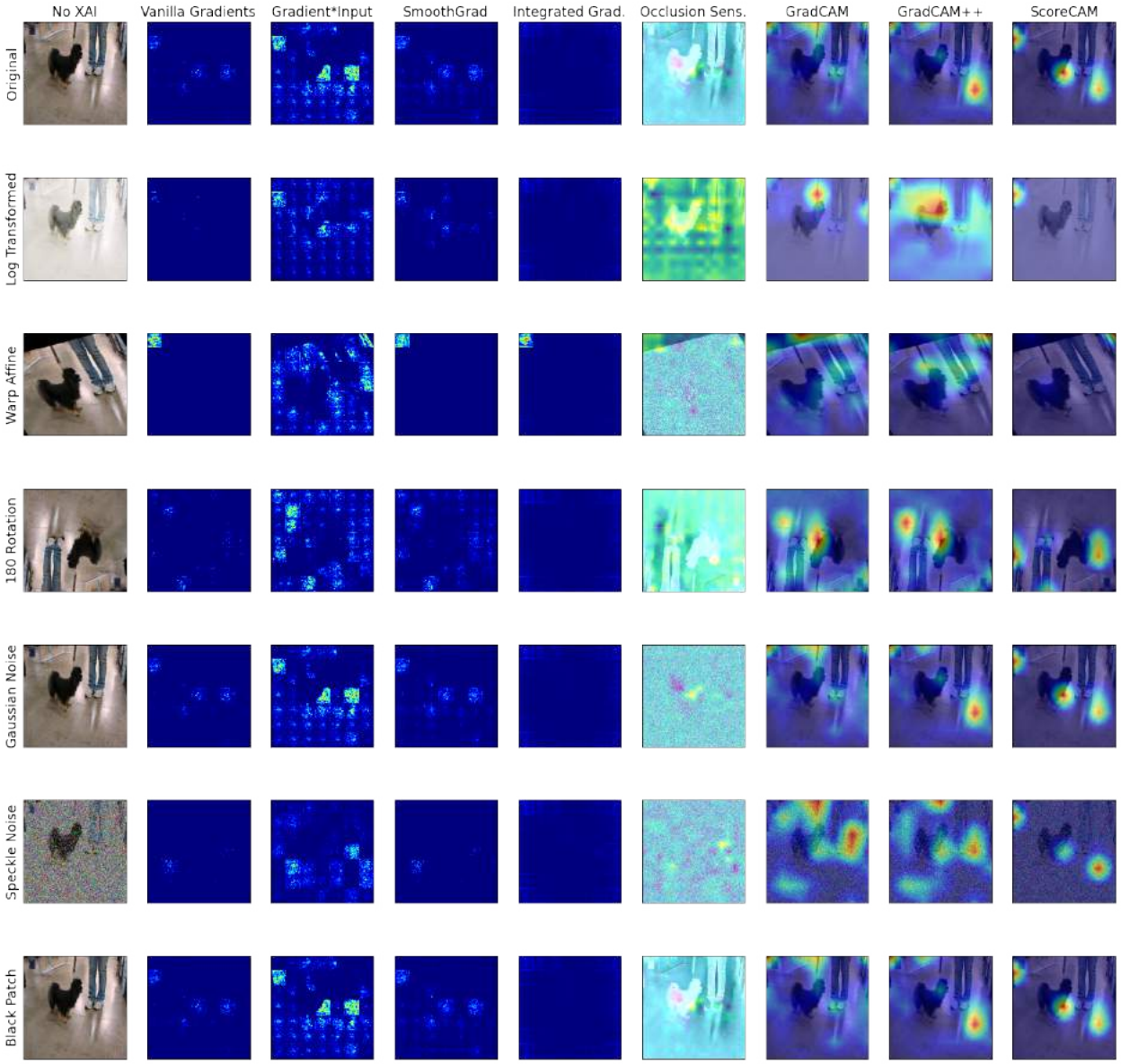


Figure 42: Visualization techniques applied to Vision Transformer (Cats & Dogs).

5 Discussion

The key difference between CNNs and transformers is the lower inductive bias of the latter, where inductive bias is understood as a design choice when creating a learning algorithm based on assumptions about the data being processed. Broadly speaking, the operation performed by convolutional neural networks is the systematic reduction of small-scale information to global information through hand-coded convolution and aggregation stages (e.g., max pooling). In the case of transformers this does not occur and they must learn this process during training, requiring a lot of data and computational resources. In fact, the ViT model used achieves its best results using the Google JFT dataset of 300 million images, available only internally to this company. Taken together, these factors limit the research to those groups that have access to large medical imaging datasets and access to extensive computational resources.

Even though the results of the networks are so disparate, we observe certain similarities: first, the effect of the dropout is evident in the performance graphs (Figuras 15, 18, 23, 26, 30, 33, 37 y 40), showing better accuracy values and losses on the validation subset. The cause of this is that learning is artificially hindered during training by eliminating a percentage of units from certain layers, while in the validation stage the neural network has its full computational capacity, performing better than during training. On the other hand, it is possible to correctly regulate the training and avoid a pronounced overfitting to the data thanks to the techniques employed.

The interpretability technique most commonly employed in transformer models used in computer vision tasks consists of simply showing the attention scores obtained by the layers that implement this mechanism, ignoring the effect of the rest of the layers, so interpretability methods designed for convolutional networks have been applied. Nevertheless, it should be noted that these approaches are not adapted to the particularities of self-attention modules, leading to results that can be confusing. For instance, in Figures 19, 27 y 34 it can be seen how the positive and negative contributions of the weights are interchanged in the gradient-based methods, except in Gradients*Inputs since they are weighted by the values of the input image, as already explained in the previous section. This result is due to the fact that these methods only take into account the output of the last layer of the model and not the partial results of each attention module or "head", so that the final image is reconstructed from the patches and the individual values obtained in each grid are indicated and not at the global level, so that the pixels in dark or black areas move in a range of tiny values and when scaling the individual contributions of these their value is multiplied exceeding the contributions of the patches with higher intensity values. The relevance values should be propagated and integrated across the grid blocks, rather than simply accumulating the individual results in the final image. In the case of the color images in Figure 42 again higher values are observed for the gradients in the dark areas of the image, for example when performing the affine transformation in the upper left corner.

As for the occlusion sensitivity method, there is no agreement on how the perturbation should be correctly applied to the input sample division. In this case it has been applied individually to each patch, resulting in a display that is not very accurate and is clearly affected by variations in the input image, especially in cases of noise addition. This is probably due to the impossibility of associating a small occlusion in one of the patches with the overall classification result.

However, in the case of MNIST (Figure 34) where a higher degree of accuracy is achieved in the results, we can observe how the CAM-based techniques focus on the characteristic zones that represent the "nine", highlighting the circular zone and the angle that it forms with the straight line. Moreover, these visualizations are practically unperturbed by input transformations. Even in the case of gradients, it is possible to identify the area in which the digit is located, in the case of Gradient*Input unambiguously and in the others by inverting the calculated values.

There are two primary factors that influence these visualizations. On the one hand, most of the studies consulted on ViT models prove that, even using transfer learning, better results are obtained using a training subset with a large amount of data, even better than using data augmentation techniques. On the other hand, since this is an n-ary classification problem, the differences between classes are more noticeable and the network can more easily discern the characteristic features of each class.

As for the visual techniques, it is quite clear that direct representations of the gradients do not provide relevant information about the model results. The same applies to the perturbation-based method. In both cases the information at the level of each patch does not provide a valid explanation, since it does not allow drawing relevant conclusions about the diagnosis provided. On the opposite, the activation maps by class do allow extrapolating conclusions and fulfill the function of accompanying the hit rate, allowing the interpretation of the model.

The poor results in the classification of mammographies by Vision Transformer are blamed on the small number of images that form them and the inability of the network to generalize. However, in view of Figure 19, especially the visualization generated by Grad-CAM, it can be seen that the network correctly identifies the region close to the lesion. Refining this localization with GradCAM++ shows that the location of this area is not as precise, but it is still located in the vicinity. It is possible that this is because the splitting of the input image allows the model to learn spatial

relationships present in the image and the abnormality in the mammography is detected, but the association between this abnormality and the label of "malignant" is not reached due to the lack of samples. In Figure 27, especially in CAM techniques, it can be seen how the network does identify certain regions of interest even by varying the input, although these regions are vaguely close to the location of the malignant mass, it can be seen that they are not random, since they remain at the edges of the same, ignoring the areas without data, except in the case of certain image transformations that significantly alter the result of the visualization.

Regarding the EfficientNet model, the results are similar to those that could be obtained with other CNNs. Not as much data is needed in the training as for the attention-based model, so the results obtained are better in terms of accuracy in all experiments, except for the classification with INbreast. Based on the images in Figure 24 it can be assumed that the problem lies in the small difference between mammographies classified as malignant and those considered benign, as well as in the excess of black areas. The gradient values are practically null, generating a minimal attribution map, even weighting them by the input values there is no clear answer. While the activation maps per class differ from each other. With Grad-CAM it appears that the warmest color zone is close to where the mass is located. However with Grad-CAM++ it does not seem to be located correctly, as there is no red zone nearby, although there are circular yellow regions quite close.

In the case of Figure 16, the most intense generated visualizations continue to concentrate on the breast relief, in no case the inner regions where the masses or calcifications that produce tumors are usually located are indicated. Considering the location of the mass in Figure 14, an area of higher intensity can be identified in the gradient-based representations, especially in the SmoothGrad result. Again, the inability to discern between network categories is demonstrated. This supports the theory of lack of convergence during training identified in the plots in Figure 15, where multiple peaks of accuracy are shown.

In contrast to the mammography cases, we have both the MNIST case in Figure 31 and the Cats&Dogs case in Figure 39 where the classification results are considerably better. In the case of the color dataset, it can be seen how the network identifies the dog figure within the image almost clearly. Gradient-based techniques, especially SmoothGrad, show the outline of the animal perfectly, indicating the regions used by the network for prediction. The same is true for the occlusion technique, indicating in lighter colors the head and leg regions. GradCAM++ also centers the warmer regions of the heat map around the head, while GradCAM generates a circular region around the dog. The analysis of the visualizations in the MNIST case is similar, the attribution maps generated by the gradient values and the occlusion technique highlight the edges of the digit, while GradCAM draws a circular map starting from the center of the digit and encompassing it almost completely. It is easy to interpret the reasoning followed by the network in both cases, visualizing the areas it identifies as relevant in the representations, identifying both the shape of the 9 and the outline of the dog and classifying them correctly.

Finally, from the point of view of the transformations applied to the input image, it can be observed that in some cases the results of the visualizations vary considerably. This effect may have the same origin as that of adversarial attacks against neural networks. This type of attack consists in the imperceptible alteration to the human eye of an input image to the network that manages to completely alter the result of the prediction made. Applying XAI techniques can help to build more robust models against this type of attacks. Observing the results obtained, it can be detected that the occlusion techniques are very sensitive to the addition of noise, providing unintelligible results. The changes presented by other techniques, however, could result in confusion on the part of the network when classifying the image produced by the transformation.

6 Conclusions and future work

In this report we have defined a workflow based on the post-hoc interpretability of neural networks, focusing on their possible application to computer-aided diagnosis systems in the field of medicine, specifically in the analysis of mammographies with the purpose of detecting possible tumor signs. For this purpose, two of the latest neural network models have been selected, one of them, with no known application by the author in breast cancer diagnosis, is based mainly on the self-attention mechanism, a novel architecture and, therefore, scarcely studied. In addition, the results of interpretability techniques that perform relevance attribution following different patterns and constructing complementary visualizations that help to understand the reason for the decision taken by the model are presented.

After experimental evaluation of the proposed architecture, it can be concluded that the results in the mammography classification task are not competitive with those obtained using other techniques documented in the area review. However, these low accuracy values are complemented by the visualizations produced, providing a possible explanation and allowing to change the direction of development in an informed manner. In this case, it may be beneficial to employ dataset annotations to crop the images and reduce them to the region of interest, thus eliminating the noise induced by the darker areas or, going to the other extreme, to employ thresholding techniques that convert the image to binary values. Some of these approaches have already been contemplated in similar works. In addition, these other systems, in order to overcome the lack of images, conveniently decompose the problem into two phases: lesion detection and classification. This way, classification is only performed on mammographies that present some type of anomaly, malignant or not. The works consulted that use this approach obtain much greater accuracy. It is to be expected that improving this task will help to obtain more explanatory visualizations of the results obtained.

The analysis performed is qualitative, based solely on visual perception and comparison of the images through observation. To complete this information, techniques could be applied on the images obtained to provide quantitative information on the accuracy of the visualizations generated by the XAI algorithms, considering them as segmentations and comparing them with the ground-truth. For example, the calculation of the Intersection Over Union metric or the average percentage of accuracy at the pixel level. They have not been included in this study due to lack of time, but it would be interesting to include these results in a future line of work in order to deepen the comparison between techniques.

By improving the input data, the ViT model seems suitable for this type of task, since the division into quadrangular samples added to the layers of attention allows the network to analyze the details present in each of these patches. In light of the results obtained on MNIST, a division into a larger number of categories could also be considered, as has been proposed in other works, for example, using the BI-RADS standard consisting of 6 different classes.

Pointing towards the explanatory visualizations generated, it seems clearly necessary to adapt most of the methods developed for convolutional neural networks to other types of architectures such as the Vision Transformer, allowing to extract information from the different attention blocks and combine them in a final attribution map, instead of simply accumulating and extracting them from the final layer. This would help to obtain clearer relevance visualizations, such as those extracted from the EfficientNet model.

It is possible to make a change in the implementation allowed by the `Python` libraries used. This modification would consist of the inclusion during training of the visualization techniques as *callbacks*, making it possible to monitor the process in real time through the `Tensorboard` tool and to establish decision making based on how the process progresses. This would imply, for example, being able to see the result of applying GradCAM to each of the training images right after they are processed by the model. Nonetheless, this line of development would increase the computational cost of the model, which is already high due to the fact that the transformer architecture is designed for natural language processing problems where the technical requirements of the system are much lower.

To summarize, interpretability is a feature of deep learning models that is increasingly desirable and sometimes mandatory in legal terms. It is useful in building confidence with the user or simply to gain a better understanding of the data and the model. The visualization of the areas of interest considered by the model in the input images allows the development of more robust models against adversarial attacks, allowing decisions to be made based on how certain input affects the learning process. The analysis demonstrates the ease of inclusion of this type of methods in existing systems and the benefits they provide by complementing the numerical results and facilitating a visual explanation of them. Combining this approach with the attention mechanism used in the transformer would enable a more widespread use of artificial intelligence in healthcare applications, taking into consideration the individual contribution of the details present in the analyzed image and allowing professionals to verify the diagnosis made quickly and directly.

References

Aibar, L., Santalla, A., López-Criado, M., González–Pérez, I., Calderón, M., Gallo, J., and Parra, J. F. (2011). Clasificación radiológica y manejo de las lesiones mamarias. *Clínica e Investigación en Ginecología y Obstetricia*, 38(4):141–149.

Al-antari, M. A., Han, S.-M., and Kim, T.-S. (2020). Evaluation of deep learning detection and classification towards computer-aided diagnosis of breast lesions in digital x-ray mammograms. *Computer Methods and Programs in Biomedicine*, 196:105–584.

Barredo Arrieta, A., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., Garcia, S., Gil-Lopez, S., Molina, D., Benjamins, R., Chatila, R., and Herrera, F. (2020). Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information Fusion*, 58:82–115.

Chattopadhay, A., Sarkar, A., Howlader, P., and Balasubramanian, V. N. (2018). Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 839–847.

Chefer, H., Gur, S., and Wolf, L. (2021). Transformer interpretability beyond attention visualization. *CVPR*, pages 782–791.

Christlein, V., Spranger, L., Seuret, M., Nicolaou, A., Král, P., and Maier, A. (2019). Deep generalized max pooling. *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1090–1096.

Dong, S., Wang, P., and Abbas, K. (2021). A survey on deep learning and its applications. *Computer Science Review*, 40:100–379.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. (2021). An image is worth 16x16 words: Transformers for image recognition at scale. *International Conference on Learning Representations*.

Eitel, F. and Ritter, K. (2019). Testing the robustness of attribution methods for convolutional neural networks in mri-based alzheimer's disease classification. *Interpretability of Machine Intelligence in Medical Image Computing and Multimodal Learning for Clinical Decision Support - Second International Workshop, iMIMIC 2019, and 9th International Workshop, ML-CDS 2019, Held in Conjunction with MICCAI 2019, Shenzhen, China, October 17, 2019, Proceedings*, 11797:3–11.

Gilpin, L. H., Bau, D., Yuan, B. Z., Bajwa, A., Specter, M. A., and Kagal, L. (2018). Explaining explanations: An overview of interpretability of machine learning. *2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA)*, pages 80–89.

Habib, G., Kiryati, N., Sklair-Levy, M., Shalmon, A., Neiman, O. H., Weidenfeld, R. F., Yagil, Y., Konen, E., and Mayer, A. (2020). Automatic breast lesion classification by joint neural analysis of mammography and ultrasound. *Multimodal Learning for Clinical Decision Support and Clinical Image-Based Procedures - 10th International Workshop, ML-CDS 2020, and 9th International Workshop, CLIP 2020, Held in Conjunction with MICCAI 2020, Lima, Peru, October 4-8, 2020, Proceedings*, 12445:125–135.

Hafiz, A. M., Parah, S. A., and Bhat, R. U. A. (2021). Attention mechanisms and deep learning for machine vision: A survey of the state of the art. *Research Square*.

Hatamizadeh, A., Yang, D., Roth, H., and Xu, D. (2021). UNETR: transformers for 3d medical image segmentation. *CoRR*, abs/2103.10504.

Hendrycks, D. and Gimpel, K. (2016). Gaussian error linear units (gelus). *arXiv: Learning*.

Khan, S., Naseer, M., Hayat, M., Zamir, S. W., Khan, F., and Shah, M. (2021). Transformers in vision: A survey. *ArXiv*, abs/2101.01169.

Lamy, J.-B., Sekar, B., Guezennec, G., Bouaud, J., and Séroussi, B. (2019). Explainable artificial intelligence for breast cancer: A visual case-based reasoning approach. *Artificial Intelligence in Medicine*, 94:42–5.

Lee, R., Gimenez, F., Hoogi, A., Miyake, K., Gorovoy, M., and Rubin, D. (2017). A curated mammography data set for use in computer-aided detection and diagnosis research. *Scientific Data*, 4:170–177.

Lenis, D., Major, D., Wimmer, M., Berg, A., Sluiter, G., and Bühler, K. (2020). Domain aware medical image classifier interpretation by counterfactual impact analysis. *Medical Image Computing and Computer Assisted Intervention – MICCAI 2020*, PP:315–325.

Liu, L., Jiang, H., He, P., Chen, W., Liu, X., Gao, J., and Han, J. (2020a). On the variance of the adaptive learning rate and beyond. *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*.

Liu, Y., Azizpour, H., Strand, F., and Smith, K. (2020b). Decoupling inherent risk and early cancer signs in image-based breast cancer risk models. *MICCAI (6)*, 12266:230–240.

Loshchilov, I. and Hutter, F. (2017). Sgdr: Stochastic gradient descent with warm restarts. *arXiv*, 1608.03983.

Loshchilov, I. and Hutter, F. (2019). Decoupled weight decay regularization. *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*.

Mehta, S., Lu, X., Weaver, D. L., Elmore, J. G., Hajishirzi, H., and Shapiro, L. G. (2020). Hatnet: An end-to-end holistic attention network for diagnosis of breast biopsy images. *CoRR*, abs/2007.13007.

Moreira, I., Amaral, I., Domingues, I., Cardoso, A., Cardoso, M., and Cardoso, J. (2011). Inbreast: Toward a full-field digital mammographic database. *Academic radiology*, 19:236–48.

Niu, Z., Zhong, G., and Yu, H. (2021). A review on the attention mechanism of deep learning. *Neurocomputing*, 452:48–62.

Oyelade, O. N. and Ezugwu, A. E. (2020). A state-of-the-art survey on deep learning methods for detection of architectural distortion from digital mammography. *IEEE Access*, 8:148644–148676.

Pérez, R., Pérez, M., Pérez, A., and Moya-Sanchez, E. U. (2020). Projectionnet and lrnet: Convolutional neural networks for mammography projections classification.

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. (2019). Grad-cam: Visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision*, 128(2):336–359.

Shin, H., Ihsani, A., Mandava, S., Sreenivas, S. T., Forster, C., Cha, J., and Initiative, A. D. N. (2020). GANBERT: generative adversarial networks with bidirectional encoder representations from transformers for MRI to PET synthesis. *CoRR*, abs/2008.04393.

Shrikumar, A., Greenside, P., and Kundaje, A. (2017). Learning important features through propagating activation differences. 70:3145–3153.

Simonyan, K., Vedaldi, A., and Zisserman, A. (2014). Deep inside convolutional networks: Visualising image classification models and saliency maps. *CoRR*, abs/1312.6034.

Singh, A., Sengupta, S., and Lakshminarayanan, V. (2020). Explainable deep learning models in medical image analysis. *Journal of Imaging*, 6(6):52.

Smilkov, D., Thorat, N., Kim, B., Viégas, F. B., and Wattenberg, M. (2017). Smoothgrad: removing noise by adding noise. *CoRR*, abs/1706.03825.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958.

Steiner, A., Kolesnikov, A., Zhai, X., Wightman, R., Uszkoreit, J., and Beyer, L. (2021). How to train your vit? data, augmentation, and regularization in vision transformers. *arXiv*, 2106.10270.

Suh, Y. J., Jung, J., and Cho, B.-J. (2020). Automated breast cancer detection in digital mammograms of various densities via deep learning. *Journal of Personalized Medicine*, 10(4):211.

Sundararajan, M., Taly, A., and Yan, Q. (2017). Axiomatic attribution for deep networks. *Proceedings of the 34th International Conference on Machine Learning*, 70:3319–3328.

Tan, M. and Le, Q. V. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. *ArXiv*, abs/1905.11946.

Tjoa, E. and Guan, C. (2020). A survey on explainable artificial intelligence (xai): Toward medical xai. *IEEE Transactions on Neural Networks and Learning Systems*, PP:1–21.

Valanarasu, J. M. J., Oza, P., Hacihaliloglu, I., and Patel, V. M. (2021). Medical transformer: Gated axial-attention for medical image segmentation. *CoRR*, abs/2102.10662.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 11:6000–6010.

Wang, H., Wang, Z., Du, M., Yang, F., Zhang, Z., Ding, S., Mardziel, P., and Hu, X. (2020). Score-cam: Score-weighted visual explanations for convolutional neural networks. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 111–119.

Yang, X. (2020). An overview of the attention mechanisms in computer vision. *Journal of Physics: Conference Series*, 1693:012173.

Zeiler, M. D. and Fergus, R. (2014). Visualizing and understanding convolutional networks. *Computer Vision – ECCV 2014*, pages 818–833.

Zhang, Y., Tiňo, P., Leonardis, A., and Tang, K. (2021). A survey on neural network interpretability. *IEEE Transactions on Emerging Topics in Computational Intelligence*, pages 1–17.

Zhou, B., Khosla, A., Lapedriza, À., Oliva, A., and Torralba, A. (2016). Learning deep features for discriminative localization. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2921–2929.

Zou, L., Yu, S., Meng, T., Zhang, Z., Liang, X., and Xie, Y. (2019). A technical review of convolutional neural network-based mammographic breast cancer diagnosis. *Comput. Math. Methods Medicine*, 2019:6509357:1–6509357:16.