Mário Alexandre Neves Gomes

Bachelor of Science in Applied Mathematics

Attention Mechanisms in the Classification of Histological Images

Dissertation plan submitted in partial fulfillment
of the requirements for the degree of

Master of Science in
Analysis and Engineering of Big Data

Adviser: Ludwig Krippahl, Assistant Professor,
NOVA University of Lisbon

FACULDADE DE
CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE NOVA DE LISBOA

June, 2022

Abstract

Recently, there has been an increase in the number of medical exams prescribed by medical doctors, not only to diagnose but also to keep track of the evolution of pathologies. In this sense, one of the medical specialties where the mentioned increase in the prescription rate has been observed is oncology. In this regard, not only to efficiently diagnose but also to monitor the evolution of the mentioned diseases, CT (Computed Tomography) scans, MRIs (Magnetic Resonance Imaging), and Biopsies are imaging techniques commonly used.

After the exams are performed and the results retrieved by the respective health professionals, their analysis and interpretation are mandatory. This process, carried out by medical experts, is usually a time-consuming and tiring task. In this sense and to reduce the workload of these experts and support decision making, the research community start proposing several computer-aided systems, whose primary goal is to efficiently distinguish between healthy images and tumoral ones. Despite the success achieved by these methodologies, it become evident that the distinction of the two mentioned image categories (healthy and not-healthy) was associated with small regions of the images, and therefore not all image regions were equally important for diagnostic purposes. In this line of thinking, attention mechanisms start being considered to highlight important regions and neglect unimportant ones, leading to more correct predictions.

In this thesis, we aim to study the impact of such mechanisms in the extraction of features from histopathological images of the epithelium from the oral cavity. In order to access the quality of the generated features for diagnostic purposes, those features were used to distinguish healthy from cancerous histopathological images.

Keywords: Machine Learning, Attention Mechanisms, Deep Learning, Histopathological Images, Diagnosis

Resumo

Recentemente, tem-se observado uma tendência crescente no número de exames médicos prescritos por médicos, no sentido de diagnosticar e acompanhar a evolução de patologias. Deste modo, uma das especialidades médicas onde a referida taxa de prescrição se assinala bastante elevada é a oncologia. No sentido de não só diagnosticar com eficácia, mas também para que a evolução das patologias seja devidamente seguida, é comum recorrer-se a técnicas de imagiologia como TACs (Tomografia Axial Computorizadas), RMs (Ressonâncias Magnéticas) ou Biópsias.

Após a recepção dos respectivos exames médicos é necessário a sua análise e interpretação pelos profissionais competentes. Este processo é frequentemente moroso e cansativo para estes profissionais. No sentido de reduzir o labor destes profissionais e apoiar a tomada de decisão, começaram a surgir na literatura diversos sistemas computacionais cujo objectivo é distinguir imagens saudáveis de imagens não-saudáveis. Apesar do sucesso alcançado por estes sistemas, rapidamente se verificou que a distinção das duas classes de imagens é dependente de pequenas regiões, neste sentido nem todas as regiões constituintes da imagem são igualmente importantes para a distinção acima indicada. Posto isto, foram considerados mecanismos de atenção no sentido de maior importância dar a porções relevantes da imagem e negligenciar menos importantes, conduzindo a previsões mais correctas.

Nesta dissertação pretende-se fazer um estudo do impacto destes mecanismos na extracção de *features* de imagens histopatológicas da mucosa oral. No sentido de avaliar a qualidade das *features* extraídas para o diagnóstico, estas são usadas por classificadores para a distinção de imagens saudáveis e cancerígenas.

Palavras-chave: Aprendizagem Automática, Mecanismos de Atenção, *Deep Learning*, Imagens Histopatólogicas, Diagnóstico

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Problem statement and motivation

Currently, the incidence of cancerous diseases is increasingly high. It is a fact that cancer is the second leading cause of death globally, accounting for an increase of 66% in the global number of deaths from 1990 to 2017 [60]. It is also of public knowledge that cancer originates from the accumulation of genomic mutations over a lifetime period in the DNA of a cell. This accumulation contributes to the development of malignancies, such as cancer, in tissue from where the cell belongs. The uncontrolled division of the mutated cell leads to the growth of the mutated tissue, often called tumoral mass [53]. To diagnose these pathologies, medical staff often make use of imaging techniques, such as CT scans and MRI, to capture what is to be believed the affected area or the tumoral mass. The resulting images are subject to experts that visually vouch for the presence or absence of cancer. Given this situation, many researchers have started to devote themselves to developing high-performance and reliable computer-aided systems to help medical staff perform such diagnose, using for that purpose biopsy images, also called histopathological images [41, 55]. Among the many steps involved in the diagnosis, usually binary classification represents a crucial stage. In this stage, the aim is to determine if the tissue captured in the image is healthy or cancerous. For these systems to perform such classification, a set of features or descriptors are extracted from the image, in a process called feature extraction, based on which the classification will be performed.

Feature extraction can be classified as automatic or hand-based. Hand-crafted features are usually created by experts and constitute a time-consuming and tedious approach, given that different features may be adopted for different tasks to produce the best classification metrics. Automatic feature extraction is a much less time-consuming approach since it is carried out by artificial intelligence models, that learn the features that produce the best classification metrics. Convolutional neural networks (CNNs), are a popular type of neural networks in image analysis due to their state-of-the-art results in several medical imaging classification tasks [25, 69].

1.2 Convolutional Neural Networks

CNNs are a type of multi-layer neural networks specifically designed to extract features from images [33]. Their ability to learn discriminative visual features from images resulted in state-of-the-art performances in several tasks, from which image classification is included. These architectures are also known by their assumptions and properties. By assuming the input images exhibit locality of pixel dependencies and stationarity of statistics [43], the model delivers local invariance and compositionality [59]. The locality of pixel dependencies or local pixel correlation means that neighboring pixels are not independent, while stationarity of statistics means that pixel values do not change over time. The local invariance is the ability to ignore the position of key structures in the image, for instance, a tissue image is classified as tumoral regardless of the localization of the tumoral area. The compositionality is an attribute that allows the network to learn high-level features from low-level features, in a hierarchical manner [59]. Another important property of CNNs is the receptive field. The receptive field can be defined as the size of the region that will be taken as input to produce a given feature. This concept is closely related to the models' performance [5], since a large receptive field enables the network to have a broader look at the input image.

Despite being proved that CNNs can deliver state-of-the-art results in a variety of tasks in the medical field, cancer classification remains a challenging task due to inter-class similarities and intra-class variability [2, 39]. Inter-class similarities are associated with the fact that the tumoral masses share visual characteristics with the tissue where it is growing, which is also composed of healthy regions. Intra-class variability is related to the heterogeneity of the tumoral masses, ranging from shapes and sizes to smoothness and texture. To produce accurate classifications deep learning models must capture these fine-grained features from relevant regions of the image. Being these regions the ones that best correlate with the target label. For instance, if a given image is associated with a 'cancerous' label, all regions where the tissue is tumoral are considered relevant for classification and healthy regions are considered irrelevant. Nevertheless, the mentioned regions usually compose a small portion of the whole image, leading to a low importance in the generation of the prediction compared with healthy regions, which usually compose a much bigger portion of the image and therefore a higher impact in the classification. This situation frequently leads to a high proportion of false negative examples, where cancerous images are labeled healthy. In order to place a higher weight on certain regions, more related to the observed label, we propose the use of attention mechanisms in the model architecture.

1.3 Attention Mechanisms

In simple terms, the concept of attention mechanism is associated with the meaning of the 'attention' word. In this regard, this class of mechanisms consists of neural networks whose purpose is to place greater importance on a given part or parts of the input, for further processing. The first attention model was used to improve the performance on machine translation tasks [8]. Nevertheless, due to the outstanding results achieved on Natural Language Processing (NLP) tasks other variants of the attention start being proposed, from which visual attention is an example.

Attention mechanisms can be globally characterized in two sets: soft-attention mechanisms and hard-attention mechanisms.

In order to better illustrate the two concepts, let us take as input to an attention module $Attention(\cdot)$ a matrix $X \in \mathbb{R}^{m \times n}$. Let us also take $Attention(X) = Y$. It is important to note that in both attention models the first step is to learn a set of weights $A \in [0, \ldots, 1]^n$, based on the input X, which is correlated with the importance of each element of X for the task at hand.

The fundamental difference between the two attention mechanisms is the way by which the weights are used to refine the input in the attention model. In soft-attention mechanisms, the refined output U is simply defined as the product of each input in X and the respected weight in A, that is, $U_{*,j} = X_{*,j} \times A_j$, in which j identifies a given column of X. Is also of great importance to mention that this type of attention is deterministic, and therefore differentiable, allowing its optimization to be carried out by back-propagation. Despite this advantage, soft-attention models can impose a computational burden depending on the network that computes the weights and the dimension of the inputs.

The hard-attention modules, contrary to soft-attention that aims to refine its input, aim to select one of the inputs of X which hopefully is related to the target class associated with X. In this sense, the computed weights A are interpreted as probabilities of each input of X to be selected. The reader needs to note that for the implementation of the hard-attention we could simply select the input of X associated with the highest probability in A, but this would not be differentiable. Therefore, complex sampling techniques must be employed. In this sense, A is considered as a parameter of a multinomial probability distribution as well as a number of trials $n = 1$. One of the drawbacks of this type of attention is the complexity associated with the optimization of the parameters of the hard-attention mechanism. Such optimization is dependent on the implementation of reinforcement learning policies that in simple terms, reward the attention for choosing the right input [73].

In this thesis, we aim to study the utility of different attention mechanisms in deep learning architectures for feature extraction tasks. This study is encouraged by the success of attention mechanisms in a variety of medical tasks (classification [74] and segmentation [35, 54]).

However, to train supervised models such as CNNs is frequently needed large amounts of labeled data. Such need imposes a problem in the medical field, since the amount of data is often scarce, being unlabeled data more widely available. In order to leverage the unlabeled data, new unsupervised approaches such as the autoencoders start being proposed.

1.4 Autoencoders and Feature Representation

The autoencoder architecture was proposed in [61], to solve the problem of "backpropagation without teachers". The architecture is another class of artificial neural networks, composed of an encoder and a decoder. The encoder compresses the input into a low-dimensional representation also known as code or embedding, through a bottleneck structure. This representation is then mapped to its original space through the decoder in a similar fashion in which the encoder operates. The autoencoders are trained in such a way that minimizes the reconstruction error, that is, the difference between input and output. The autoencoder, as mentioned above, learns a new compressed set of features representing the input, having the capability of capturing complex relationships that might take place [9]. As such, it can be used as a powerful feature extractor for dimensionality reduction. Recent studies have shown that the weights of the encoder can also be used to provide an initialization for the weights of a supervised model, avoiding a random initialization of the classifier. Several authors have shown that this approach leads to small but consistent improvements in classification performance [49, 75].

Autoencoders have achieved great results in a variety of scenarios when associated with other learning models, by not only successfully extracting useful and robust features that make tasks like classification much easier but also avoid that the models get stuck in local mínima [38, 64, 75, 79].

1.5 Objectives and expected contributions

Recently, attention mechanisms have been extensively used in state-of-the-art architectures in Natural Language Processing (NLP) tasks. The breakthrough and success of these mechanisms in the mentioned field motivated their use in other fields of deep learning such as Computer Vision.

The main goal of this thesis is to assess the utility of these mechanisms in feature extraction tasks. For that, in an initial stage we have developed an attention-based convolutional autoencoder (CAE) using for training purposes histopathological images of the oral cavity. In this first step, the attention-based CAE was trained on random patches extracted from those images. In a second and final stage, a SVM classifier was trained on the representations generated from the encoder to predict the diagnosis of each image concerning the presence or absence of oral squamous cell carcinoma.

In this manner, this work to has the following contributions:

- A python package for the unsupervised feature extraction of histopathological images through the use of attention mechanisms;

- A python package for the binary classification of histopathological images, for the presence of oral squamous cell carcinoma, through the use of extracted features

Chapter 2

State of the art

2.1 Handcrafted-based Feature Extraction

In former times, cancer diagnosis was held by pathologists. In that times, these professionals would examine histopathological images of biopsy samples and judge their nature - cancerous or healthy - based on their experience. Despite their knowledge on the mentioned topic, their judgement is subjective, leading to considerable variability depending on the case at hand. In order to reduce the subjectivity and improve reliability of cancer diagnosis, computational tools for image classification start being proposed. For these computational tools to classify in a reliable way such images, the determination of visual features, that best not only represent the images but also discriminate between healthy and cancerous ones, is of utmost importance. In this sense, numerous studies in the literature propose not only feature extraction techniques that aim to obtain such features but also preprocessing techniques to improve image quality. Feature extraction methods aim to characterize biological images by taking advantage of properties such as shape or texture in specific ways. Shape features describe the geometry of a given region and are commonly considered for problems of cancer diagnosis [1]. Given that in contrast to healthy cells, cancer cells exhibit abnormal and irregular shapes and tend to be much larger or smaller compared to healthy cells [16]. Texture features aim to capture spatial relationships between neighboring pixels and play an important role when cancer provokes tissue heterogeneity [65]. In the table 2.1 we present some extraction techniques associated to different physical properties of the images.

Image Property	Extracted Feature	Description
Shape	Compactness	Ratio of the square of the perimeter to the area
Shape	Major Axis	Length of the longest chord that goes throught the center
Shape	Perimeter	Sum of the distances between every consecutive boundary point
Shape	Sphericity	Measure of the degree of roundness of a volumne
Texture	Contrast	Aggregation of the difference in the luminance of sub-images
Texture	Smoothness	Aggregation of regularity of pixel intensity across sub-images

Table 2.1: Feature extraction methods and respective physical properties.

In [15] a study is conducted for classifying histological breast cancer images. In the proposed architecture, the images are subjected to preprocessing techniques for nuclei detection and image quality improvement. From the detected nucleus, a vector of features is considered. For the construction of this vector, shape features were considered - for each detected nuclei measures such as area and symmetry were extracted, as well as texture features - amount of nucleus and DNA strands. The feature vectors are fed to a classifier that aims to distinguish between cancerous and healthy tissue images.

Although available in the literature a vast selection of feature extraction techniques, it may be unclear which properties of the image should be considered and what physical features should be used to produce the best metrics. This is a common issue and constitutes a central problem in handcrafted feature extraction pipelines, since the type of feature to extract depends heavily on the task at hand. For instances, shape features are useful in the diagnose of lung cancer and are used to predict the response from patients to treatment [65], while texture features are good predictors of lung cancer outcomes [23]. Given this situation, it is a common procedure to consider different feature extractors associated to different image properties. Nevertheless this leads to a very high-dimensional feature space. The high number of dimensions associated to the feature space leads to a phenomenon known as curse of dimensionality. The curse of dimensionality not only makes it difficult to perform data analysis but also increases the chance of problems such as multicollinearity and numerical instability associated to the parameters of a model. Multicollinearity describes a situation where, at least, 2 features are significantly correlated. Such problems during model training can translate to overfitting, training instability and prevent interpretation and statistical inference on the parameters of the model. To address this drawback, the most informative features must be identified, usually through statistical hypothesis testing. This step takes the designation of feature selection or feature reduction and is frequently a part of the image analysis pipeline. Huang and Lai [32] to classify liver biopsy images considered a fourteen-dimensional feature space composed by size, shape, texture and intensity features. For classification purposes, a tree-based classifier was considered. In each node of the mentioned tree, a feature selection procedure took place in order to prevent irrelevant features from interfering in the decision process and reduce the dimension of the feature space. In further analysis, the authors inferred that the performance of the classifier with feature selection at each decision node was better than the absence of it. This conclusion confirms that not all features are equally important for the classification of a given image but also that the same feature may be of utmost importancy in a given scenario but may be noise in other scenarios. Despite the quality of the outputs produced by the mentioned methods in small to medium datasets, the increase in the prescription of medical exams led to an increase in size of medical image repositories. Such growth imposes new challenges to the application of the mentioned methodologies such as the preservation of computational efficiency, handling intra-class and inter-class image variability while keeping the good evaluation metrics previously achieved. On the other hand, the images need to be categorized and stored accordingly for future retrieval. For this categorization, and depending of the task at hand, a given image may exhibit more than a single region of interest. Under this assumption, paradigms such as the bag of visual words (BoVW) start being used for image representation and classification [6, 68, 78]. In this approach, for each image a set of regions of interest (RoIs) are identified and subjected to feature extraction techniques resulting in a set of feature vectors. In a second stage, clustering is performed over the feature vectors associated to each RoI of each image, where each cluster corresponds to a visual word which will be used to generate a representation for each image. In this sense, each image starts being represented by a histogram in which the x-axis is related to the previously determined visual words and y-axis is the count of the RoIs whose feature vector lies in the cluster of each visual word. For classification purposes, the histograms are further fed to a classifier.

Despite the improvement brought by the BoVW over other methods, certain challenges remain unaddressed. For instances, the BoVW may allow a given image to be characterized by a set key points locations from which feature descriptors will be extracted but which feature extractors or physical properties to consider remain a challenging task besides the identification of RoIs being heavily dependent on the labour of health professionals. On the other hand, Lai and Deng [46] demonstrated that as the number of training examples increases, the accuracy of models trained on handcrafted features is hardly influenced. Thus, demonstrating that handcrafted methods not only do not adapt to large datasets but also tend to have a poor generalization power at certain tasks. Also the optimization process of the image classification pipeline is highly manual in the preprocessing, feature extraction and selection steps. Image quality improvement is often considered in the preprocessing phase. Such improvement is done through methods that very often depend on the manual optimization of threshold values. Given the drawbacks imposed by handcrafted methods it is necessary to find new techniques that can simultaneously identify the most informative features while also not requiring manual optimizable.

2.2 Deep Learning-based Feature Extraction

2.2.1 Deep Supervised Learning

To address the mentioned issues such as the slow optimization of handcrafted methods and the amount of work of medical staff, automatic computer-aided systems for image analysis start being proposed. From this class of systems deep neural networks (DNNs) occupy a central position, especially convolutional neural networks (CNNs). CNNs are widely used in medical image classification tasks and have shown significant performances since 2012 [13, 19, 57, 80]. This class of models are overtaking traditional methods given not only their abiliity to perform feature extraction in an automatic and trainable manner but also mitigate the chances of a incorrect diagnosis due to inexperience or fatigue from the medical staff. In addition, CNNs can learn a hierarchy of features in a sequential manner of increasing complexity starting by texture and edge and ending in complex structures like tumours or parts of a cell, allowing them to compete with medical experts.

Lai and Deng in [46] compare the performance of SVM classifier, trained in handcrafted features, with a CNN model in the classification of histopathological images. The authors demonstrate deep learning model outperforms the shallow classifier in 2 benchmark datasets, proving that the CNN are capable of learning richer features that translates to a better performance. It is also of interest to observe that unlike traditional machine learning methods, increasing the number of training examples leads to a increase of performance of DNNs. This observation suggests that deep learning models have a powerful generalization ability while the increase in the amount of training data hradly influences the performance of the SVM classifier.

The ability of CNNs to automatically learn features from the underlying data in an automatic way motivated the scientific community to propose deep learning architectures in order to address several challenges in the medical field. Anthimopoulos et al. [4] proposed a CNN architecture for the classification of lung CT scans, which outperformed state-of-the-art methods using handcrafted features.

It is also crucial to note that the complexity of the problem being addressed is closely related with the depth of the architecture, in other words, the harder the challenge the more layers the model will need to reach a good performance, which also translates into a need of more training data. Frequently this imposes a challenge in the use of such models, considering that medical images are hard to collect and their labelling is a time-consuming task not to mention that the collection itself raises data privacy concerns. To tackle more complex problems, having medium to little data available, Yadav and Jadhav [76] suggested the use of transfer learning. This technique can be summarized in 2 steps - the first, would be the transfer of the weights associated with the initial layers of a previously trained model to our model, and the second, use the small amount of labeled data available in order to train the last layers of our model, associated with high-level features, given that they are specific for the problem at hand. The logic behind this tecnhique is that low to medium level features such as edges or color can not only be useful for the task they were learnt but also to other simillar tasks while high-level features are more connected to the task at hand. Kermany et al. [40] demonstrated the ability of transfer learning when handling problems in which a little amount of labeled data is available. In their study, a transfer learning system was employed to classify optical coherence tomography (OCT) images. In their work, the weights from the initial layers of a pre-trained model in the ImageNet dataset [17] were used for the mentioned classification. The authors reported a performance compared with six human experts having little labeled data available but also concluded that the performance of the trained model would be inferior to one trained from the scratch in OCT images. Since in training a model from scratch all the weights would be optimized to detect OCT features. One of the aspects that influences the success of transfer learning is the choice of the pre-trained model from which the parameters will be extracted.

In this sense, data augmentation strategies start being proposed in the literature as a way of increasing the number of training images [7, 22, 47, 66, 76] so that every weight of the CNNs could be optimized for the task they are going to perform while also addressing problems of class unbalacing. This kind of problems arises when the amount of training examples associated with different categories of the target is different. In [66] the authors studied the benefits of classic data augmentation in handling a small and unbalaced dataset for the detection of skin cancer. They hypothesized that such methodologies could increase the CNN invariance to conditions - such as reflections, rotations, horizontal and vertical flips - and biological patterns like color. In this respect, the authors considered 2 experiments where the set of augmentation techniques was applied but maintaining the original ratio of imabalacing between the 2 classes and other where only the images belonging to the less observed class were subjected to data augmentation to equal the amount of training examples between the 2 classes. The intrepretation of the metrics generated by both experiments led to the conclusion that the use of data augmentations in order to make the dataset balanced produced the best results given that the network is less biased to classify the majority of the instances as belonging to the most present class. However synthesized images must look alike real images that could be found in the "real-world"if more data was collected but it is also important that the applied transformations reflect variations that will help the model better generalize to unseen images. Depending on the problem at hand the task of performing data augmentation may be more or less challenging, and in some scenarios there is a need for the synthetic images to be generated by a DNN such as a generative adversarial network (GAN). This form of data augmntation is explored by Fri-Adar et al. [22] for the classification of liver lesions in order to deal with data scarcity. In their work a comparative study between classical augmention and artificial augmentation (GANs) is carried out. The standard classic augmentions - such as translation, rotation and flipping - were used and on the other hand, a GAN was adopted for each type of lesion.

The fundamental ideia behind a GAN is that it learns the data distribution associated to set of images to further generate new images from the learnt distribution. This is done through the use of 2 CNNs, where one works as a generator that takes a random vector, usually from a standard uniform or gaussian distributions, and generates an image that follows the distribution this network has learned so far. The discriminator takes as input the generated image and other image from the distribution the generator is supposed to learn, and determines if the 2 images come from the same distribution. The authors concluded that classical augmentation improved performance up to a certain level, above which performance would not improve where in data augmention performed by the GANs, the accuracy of the classification system improved in 7,1% over using classic augmentation. Fri-Adar et al. justified this improvement saying that the GANs can learn the underlying distribution associated to each target class and therefore add variability to the input dataset that in turn led to a better performance. It is also pertinent to note that for the training of each GAN, a maximum number of sixty-five images were used. This indicates that GANs can capture the underlying data distribution even when they have, for that purpose, few training examples.

Although CNNs have emerged as a state-of-the-art approach in image problems and despite techniques such as transfer learning or data augmentation, this class of DNNs still requires a substantial amount of labeled data depending on the task at hand. Such annotations are awfully expensive to obtain given to be product of the labor of experts. On the other hand, unlabeled data is available in much higher quantity compared to labeled data and this led researchers to start proposing systems that can feed on this type of data, namely semi-supervised deep neural networks.

2.2.2 Deep Semi-Supervised Learning

From deep unsupervised models, autoencoders occupy a central position and have been extensively used in medical image classification systems [18, 21, 37, 62]. Given the encoder to be a bottleneck, autoencoders are forced to learn a meaningful and efficient representation of the input to reduce the reconstruction error. The properties of such structure are dependent on the type of autoencoders. The stacked sparse autoencoders (SSAEs), the class of autoencoders most frequently used, imposes sparseness constraints on all hidden nodes as well as a sparse penalty term in the loss function which forces all entries of the code vector to be close to zero. In [21], the authors proposed to evaluate the performance of a SSAE architecture in the detection of chronic kidney disease and cervical cancer as well as to determine the probability of a patient to develop coronary heart disease within a 10-year frame. The authors advocate that not only the adopted autoencoders learned an input representation that leads to optimal classification results but also that they are suitable for situations where data imbalancing is observed given to not rely on labeled data. In this work the authors start by training the respective autoencoder and in a second stage the learned embeddings would be fed to a softmax classifier which in turns generates the prediction probabilities. It is also important to note that decreasing the dimension of the input also translates into a reduction in the number of parameters of the softmax classifier and the probability of overfitting. It is trivial that if we decreasse the size of the input a less deep network, and threfore with less parameters, is required to classify the inputs. The architecture proposed in [21] - the encoder posteriorly associated with a softmax layer - is widely used since it presents good results and little amount of labeled data is required. In a first phase an autoencoder is trained, being after the encoder associated to a softmax layer, that performs classification based on the features extracted by the encoder.

The authors of [62] proposed to identify the type of image views - short axis (SAX) or long axis (LAX) - in MRI images. In their work, the authors aimed to study the extraction power of the autoencoder, considering for that purpose the autoencoder and a softmax classifier. The proposed schema achieved a minimum in accuracy of 91,98% for the classification of the image types, demonstrating that autoencoders are a powerful and simple tool for nonlinear representation of input images.

Despite the mentioned architecture to be very versatile and to display good evaluation metrics, more complex problems require equally complex architectures. For the diagnosis of Parkinson's disease Kadam and Jadhav propose an ensemble method based on sparse autoencoders [36] that outperformed SSAEs and softmax classifiers, by achieving more than 90% in accuracy. For the building of the architecture 2 autoencoders were considered. In this sense, the training data is first fed to the autoencoder SAE1 yielding the FS1 representation, which in turn is fed to the second autoencoder, yielding the FS2 representation. The concatenation of the 2 representations FS1 and FS2 generates a third representation FS3. Each of the 3 representations is fed to 3 different softmax classifiers for prediction generation.

Despite the demonstrated performance of SSAEs in performing feature extraction tasks, when handling image data such extraction methods do not consider their associated spatial dependencies given that images need to be flattened to fit these models. In this sense, an autoencoder that could take in consideration the correlation between neighboring pixel values would be of great usage. In the light of this, Guo et al. [27] proposed the use of convolutional autoencoders (CAEs) to perform feature extraction in images, preserving spatial and locality properties. The authors advocate that CAEs are superior to SSAEs in learning features from unlabeled images given that they incorporate spatial relationships between pixels through the use of convolutional layers. In [52] Naderan and Zaychenko compared the performance of a fine-tuned convolutional autoencoder with a CNN model and 3 fine-tuned pre-trained CNN models on the ImageNet dataset [17] for the diagnosis of breast cancer. Their experiments showed that the convolutional autoencoder achieved better results, and compared to the other methods the training time as well as the amount of parameters was lower, which also translates to less likelihood of overftting. Besides Naderan and Zaychenko, Chen et al. took advantage of a convolutional autoencoder for the classification of lung CT scans [12]. The authors conducted a study to compare the performance of the adopted convolutional autoencoder with a stacked sparse autoencoder, concluding that the convolutional autoencoder is superior than the other one when considering image data (SSAE : [accuracy = 0.77, AUC = 0.83], CAE : [accuracy = 0.95, AUC = 0.98]).

The use of autoencoders is a powerful tool when having little labeled data since it can learn useful representations from the input data. Besides this, usual problems in classification such as data imbalacing do not arise given the absence of labels. However, along with these and other advantages this class of DNNs also exhibits some drawbacks such as the interpretability of the models since the importance assigned to each region for the generation of the code is unknown. Because autoencoders are forced to prioritize which aspects of the input should be considered for the code, so that a good reconstruction can be obtained, the autoencoder might neglect small parts of the input despite their relevance to a given problem. For instances, in the diagnosis of cancer based in histopathological images, cancerous regions often compose a small region of the images, being the majority of the image associated to healthy regions. The fact that the healthy regions occupy more area translates into a higher impact in the reconstruction error and a higher presence of these regions in the code. In this sense, despite cancerous regions containing the most relevant information, their presence in the code tends to be much smaller. In order for relevant information to be considered despite taking a small portion of the input attention mechanisms are a possible solution.

2.2.3 Attention Mechanisms

Attention mechanisms aim to resemble the way pathologists analyze images. These health professionals start by identifying abnormality regions and then not only analyze them but also the respected surrounding areas. Having this procedure in mind is obvious that not all regions of the image contribute evenly for the diagnosis. In this sense, attention mechanisms aim to give more importancy to relevant regions for the diagnostic and neglect not important ones.

In [10], BenTaieb and Hamarnesh developed an attention-based model for cancer prediction. The authors advocate that given the dimensions of the images, it was unpractical to directly feed them to a DNN, given that it would imply a huge computational cost. Image resizing and cropping was also considered, but both would led to information loss. In this sense the developed system is able to classify a given slide image based on a limited number of glimpses to specific locations of the image. The generation of the location is a sequential process carried by the attention module, where each selected region is classified and the computed probabilities were average to generate a diagnosis. Regardless of the promising results achieved by the proposed model, in the presence of micro-metastases the model is unable to correctly classify the region returned by the (hard-)attention mechanism. Further analysis shows that the misclassification is associated with the high ratio of healthy to cancerous tissue in the extracted patches that is, the extracted pacth not only contains tumoral tissue but most of the times non-tumoral one. To solve this limitation of the proposed methodology, the extracted patches should be subject to a sub-system, namely a soft-attention model, in order to highlight only the pixels associated to the lesion.

This was the approach taken by Xu et al. for the diagnosis of breast cancer [72]. In the proposed system the authors considered a hard-attention mechanism to select the regions related with the abnormality part, in turn the extracted patch is subjected to the soft-attention mechanism. The soft-attention mechanism is constituted by a trunk and mask branch, where the trunk branch aim to extract features out of the extracted patch and the mask branch learns a mask in the unit interval that aims to highlight the most relevant pixels for the mention classification. The proposed architecture outperfoms state-of-the-art architectures and the authors link this to the use of the soft-attention mechanism, without which the classification accuracy would drop around 10%. Xu et al. consider the soft-attention an essential part of the proposed model since it encourages the network to neglect unnecessary image features that would led to misclassifications. Further efforts on the optimization of the mention system led to publication of [71]. In [71] the authors claim that not only hard-attention mechanisms usually take a long time to converge but also that they might select patches that even though are related with the lesion region do not improve the discriminatory power of the network. In this sense, the hard-attention module not only has to select the, hopefully most related patches to the abnormality, but also to assess their relevance for training. In this work, Xu et al. also conducted a parallel study to evaluate the importancy of the hard-attention and soft-attention mechanisms. By removing the hard-attention mechanism, the authors resized the images to 112x112 resolution and detect a maximum reduction of accuracy in 2,4 percentual points. In removing the soft-attention mechanism the authors conducted 2 separated experiments - replacing the mention module with a pretrained arechitecture having 18 layers and other having 50 layers - being the model with 18 layers the one that generated better results. The replacemente of the soft-attention mechanism with the pretrained model of 18 layers resulted in the maximum reduction of accuracy in 3,7 percentual points. From this perspective it is clear that soft-attention mechanism performs a more important role compared with hard-attention mechanisms.

However, several disadvantages are pointed to hard-attention mechanisms. This class of attention mechanisms displays problems of temporal inefficiency due to their time to convergence and being a non-diferentiable method, cannot be optimized through backpropagation [63, 72]. In this sense, the simultaneous optimization of the hard-attention model with the other weights of the network constitutes a problematic task. Besides this, also Xu et. al arrived to the conclusion that the soft-attention model is of greater importance compared to the hard-attention model. These findings led the scientific community to propose DNNs where only the soft-attention module is present. In the sense, the rest of this chapter will be focused on the analysis of soft-attention modules and respected characteristics.

In the 2 above works of Xu et al., the architecture of the soft-attention module was inspired by the work of Wang et al. [67]. In that work, a soft-attention mechanism, that incorporates the concept of residual learning and bottom-up top-down architectures, is proposed. The residual learning framework was proposed by [28] in order to facilitate the training of very deep neural networks. Such networks, due to their depth, are more likely to suffer from vanishing gradient problems. This problem affects particularly the early layers since the gradients are too small to perform a meaningful update on the weights. In order to overcome this situation, He et al. propose to add shortcut connections that perform identity mapping. In this line of thinking, the inputs are transformed through a series of layers (residual unit or block) into an intermediate output that is then summed, in a element-wise manner, with the inputs. The addition of the shortcut connections has proven to mitigate the vanishing gradient problem while neither adding extra parameters nor computational complexity.

In this architecture, the trunk branch solely composed by residual units allows more complex features to be computed without the cost of vanishing gradient problems and saturation of performance metrics. Despite the importancy given to the trunk branch, the success of the attention module lies in the architecture of the mask branch which is a bottom-up top-down architecture that weighs each pixel that composes the output of the trunk branch according to their relevance.

The mask branch can be decomposed in a bottom-up and top-down structures. The purpose of the bottom-up structure is to collect global information from the whole image, which is done by several pooling layers and residual modules. The residual modules are placed between pooling layers and aim to extract features during the downsampling of the inputs. After reaching the lowest resolution, the global information retained is expanded by a symmetrical top-down architecture. The top-down segment was built similarly compared with the bottom-up segment. Thus, being formed interchangeably by bilinear interpolation blocks and residual modules. Similar to the bottom-up segment, the residual modules aim to extract features during the upsampling process. The mask branch ends with 2 consecutive 1×1 convolution layers and a sigmoid activation that aim to learn the importancy of each pixel of the trunk mask output. It is also relevant to note that each trunk branch is associated with the respected mask branch, since each mask branch aims to improve the features learnt from the trunk branch rather than compute the best features for the problem at hand. To evaluate the performance of residual attention module, the authors conducted 2 separated experiments. In one of the experiments the authors compared a pre-trained ResNeXt-101 and a pre-trained Inception-ResNet-101 with a AttentionNeXt-56 and a AttentionInception-56. The AttentionNeXt-56 results from the intercalation of attention modules with ResNeXt and Inception basic blocks. While the AttentionNeXt-56 achieves competitive performance compared with the associated pre-trained model it has significantly fewer parameters. In the other hand, the

AttentionInception-56 outperforms the Inception-ResNet-101. In the other experiment the performance of an Attention-92 was compared the ResNet-200 and Inception-ResNet-v2, which are state-of-the-art algorithms in the ImageNet dataset. Being the Attention-92 composed of 6 attention modules, separated by residual units, it outperformed both state-of-the-art CNNs by a large margin.

The residual attention module composed by the trunk and mask branch is aimed at very deep CNNs, given that only very deep architectures tend to suffer from vanishing or exploding gradient problems which makes them difficult to train. In each attention module a set of new features is computed and weighted in such a way that the most discriminative features are emphasized. In this regard, the mentioned residual attention module will be object of comparison held on this thesis. This choice is not only linked to the state-of-the-art results achieved in classification and segmentation tasks [35, 71] but also with the implementation logic. The fact that a 3-dimensional attention map is computed allows for each pixel that composes the output of the trunk branch to be calibrated by its respected weight. In second analysis, the architecture of the attention module relies on residual learning, which reduces the probability of vanishing or exploding gradient problems despite the depth of the model where the module is inserted.

In contrast with this kind of attention module, others aim to work as gating mechanism. That is, they do not aim to compute a new set of features like the trunk-mask branch but to weight or calibrate the already computed ones.

A example of such attention model is the squeeze-and-excitation block proposed by Hu, Shen and Sun in [30]. In this work, the authors aim to build a block that weights each channel of its input by modelling the dependencies between the mantioned channels. Then, each output channel equals the input channel multiplied by the respect weight. The architecture of the block can be decomposed in a squeeze operation following an excitation operation. The goal of the squeeze operation is to build a global channel descriptor, which aims to characterize each channel in a given way. For the squeeze operation the authors opted for a global average pooling. Being computed the global channel descriptor this information is then subjected to the excitation operation which aims to learn a set of weights that will re-calibrate each channel of the input. The excitation operation is carrried by a 2-layer perceptron model that forms a bottleneck. Being the global channel descriptor a vector of size C, it is initially mapped to a vector of size C/r (where r is a consant reduction ratio). The bottleneck was adopted to reduce the computational cost of the operation while learn channel dependency features, through the application of a ReLU activation. In a second step, the compressed vector is again mapped to its original size C and subjected to a sigmoid activation in order to compute the coefficients associated to each channel. In a final step, each feature map/channel from the input is scaled by the respected coefficient with the purpose of salient the most important channels.

The simplistic and lightweight nature of th squeeze-and-excitation block also known as SE-block led researchers to start proposing deep learning architectures that took advantage of the mention block. In [24], Gong et al. proposed 2 3D CNN equipped with residual SE-blocks to perform pulmonary nodule localization and reduction of false positives in early lung cancer treatment, respectively. In this sense, firstly pulmonary CT images are fed to the first 3D symmetrical CNN so that nodule candidates can be identified. Given the nature of the problem, it is frequent that CNNs confuse pulmonary nodules with structures such as pulmonary vessels or bronchus, and aiming to reduce the false positive ratio the authors proposed a second 3D CNN to perform such distinction. The intention of the authors in the adoption of the residual SE-block was to reduce the impact of less informative feature maps in the pipeline while emphasizing the most informative ones. The proposed framework exhibited state-of-the-art performance compared with other architectures, which according with Gong et al. is partly due to the SE-block that not only makes the deep neural network easier to optimize but also less vulnerable noisy features. Besides Gong et al., also Yan et al. in [77] took advantage of the SE-block to classify thoracic diseases and locate suspicious lesion regions based on x-ray images. The authors opted a pre-trained model, in which SE-blocks were added afterwards. Yan et al. advocate that multiclass classification can be a difficult task given that the contrast difference between lesion regions of different types of diseases can hardly be distinguishable. In that way, the authors claim that the use gating mechanism associated with the classification pipeline would provide useful information for disease classification. During this study, the authors concluded that without the SE-blocks the average AUC among all respiratory pathologies would decrease in 0,0023 units of measure proving the positive impact of such mechanism.

The SE-block was designed to highlight the channels of the input that might contain useful information in order to discriminate between the target classes of the problem at hand. In some problems the localization of the most informative channels alone can lead to sub-optimal results due to the presence of irrelevant pixels in those channels. In situations where the ground truth is only related with partial regions of an image like nodule detection, instead of considering each region of the image equally, spatial attention mechanisms attempt to pay more attention to certain regions [11]. The idea of not only calibrate the channel dimension but also the spatial dimension motived Woo et al. in proposing the Convolutional Block Attention Module (CBAM) [70]. The goal of the CBAM, like the SE-block, is not to compute a new set of features but to refine the existing ones in both channel and spatial-wise manner.

The channel attention module developed by Woo et al. is similar with the SE-block in the sense that the first step is to compute a global channel descriptor vector. Unlike Hu et al. that used the global average pooling to generate the channel-wise statistics, Woo et al. argue that besides global average pooling also global max pooling gathers important information that can infer finer channel-wise attention weights. Thus, the authors use both feature descriptors simultaneously. Both descriptors serve as input to a 2-layer perceptron bottleneck structure to extract 2 attention score vectors, that are later added together and subjected to a sigmoid function to generate the final channel attention vector. The channels of the input are calibrated by this attention vector and the product is subjected to the spatial attention, in which both max pooling and average pooling operations are performed for each spatial position and along the channel axis. Both spatial descriptors are concatenated and unified to a unique bidimensional feature map. Such unification is done through one convolutional layer with 1 filter of size 7×7. The motivation associated to the use of the convolution lies in the fact that we wish to compute an attention spatial map to weight each spatial position of the input. It is of the utmost importance that this operation is learnable since the prevalence of both max-pooled and average-pooled features may differ in different points of the network. In the determination of the filter size of the convolution the authors noted the adoption of increasingly large filters consistently generated better accuracy. This makes sense since larger receptive fields imply a broader view for the convolutional filter, being the later more capable of performing the weighting of the spatial positions. To evaluate the CBAM block, it was added to several ImageNet pre-trained state-of-te-art models - such as ResNet-101, ResNeXt-101 or the MobileNet - in order to compare with the respected baselines. The networks with CBAM consistently and significantly outperformed the respected baseline, showing the power of the proposed approach. The block depicted by Woo et al. has been the basis of many deep learning systems, given to be a lightweight form of refinement of the input in both the channel and spatial dimensions. Upadhyay and Banerjee proposed a framework for medical classification called Class Specific Convolutional Coders (CSCC), where they aim to learn feature from a small amount of labelled images. In its essence they considered C encoder-decoder architectures, being C the number of classes associated to the target variable. Each encoder-decoder architecture is composed by 3 sub-modules - an encoder, an decoder and a classification layer. In this regard, all images associated to the target class C_i are used to train the C_ith encoder-decoder architecture. Each image is then fed to the input to retrieve the learnt representation, which in turn is fed to the classifier to predict if the image is from the C_i target class and to the decoder to retrieve the reconstructed image. For inference purposes, a given image is propagated through all encoders and the representations are concatenated and fed to a fully connected classifier to infer the most probable target class. The authors reported that considering the high within-class variability, in the encoderes, decoderes and classifiers were inserted a CBAM module to highlight the most discriminative features. The framework was tested in 2 datasets - one concerned with the detection of tuberculosis in pulmonary chest x-ray images and the other with the detection of a invasive ductal carcinoma in histopathological images. The accuracy of the system with and without the CBAM was studied as a function of the number of training examples, in which the association CSCC + CBAM consistenly outperforms the CSCC alone in both datasets, showing that the incorporation of the mentioned module further improves the framework.

We finish the state-of-the-art chapter declaring that, besides the TMB block, the CBAM block will be an object of analysis in this thesis. This decision is linked to their performance in both CNNs and AEs, while also being a computationally efficient methodology. It is also our opinion that both attention blocks contrast with each other, which makes their comparison interesting for the purposes of this work. For instance, the calculation of the attention weights in TMB blocks simultaneously takes into account the channel and spatial dimensions while the CBAM block weights each dimension separately. From another angle, the CBAM performs calibration in a lightweight manner requiring few parameters for that matter while the TMB performs a more computationally costly weighting by taking advantage of residual blocks and other operations.

Chapter 3

Dataset Description

The used dataset is composed of 696 images of size 2048×1536 taken from the oral epithelium and extracted from 230 patients with a 400x magnification. Those images are naturally divided into two sets - a healthy and a cancerous one. The healthy set is composed of 201 images in which the oral epithelium is healthy, being the other set characterized by the images in which some extent of tumoral tissue is observed among the healthy ones. The tumoral tissue is associated with the Oral Squamous Cell Carcinoma (OSCC), the most common malignant neoplasm affecting the oral cavity.

The histopathological images (or biopsy slides) were obtained through a Leica DM 750 microscope, model ICC50 HD connected to the camera., being previously treated with Hematoxyline and Eosin (H&E) to enhance the contrast between the cellular components. It is also important to mention that the biopsy slides were collected from two reputed healthcare service institutions - Ayursundra Healthcare Pvt. Ltd and Dr. B. Borooah Cancer Institute.

Chapter 4

Experimental Setup

In this chapter, we aim to detail the experimental work that was carried out during the making of this thesis. During the reading of this chapter, the reader will understand the motivation for certain decisions regarding, for example, parametrical choices and architectural options. All code was developed with the support of Keras API of Tensorflow 2.6.0.

4.1 Global Experimental Scheme

As stated in the 1 the fundamental goal of this thesis is to determine if the use of attention mechanisms associated with autoencoder architectures leads to the generation of richer image representations, which in turn translates to higher classification performance metrics.

To answer this question we first need to adopt several autoencoder architectures. Each architecture will be associated with three autoencoder models, in which two are equipped with two different attention mechanisms, CBAM and TMB, and a third, named standard, that does not hold any kind of attention mechanism.

In a second phase and after obtaining the representations of each image, these representations will feed an SVM classifier that aims to classify each image, using for that purpose its representation, as cancerous or healthy.

4.2 Train, Validation and Test Sets

The entire dataset was split into train, validation, and test sets considering a stratified split based on the class of the target associated with each image. Although the first stage of training is unsupervised, the second stage is supervised. Due to the second stage of the training to be supervised and the dataset being unbalanced, stratified random sampling was used to generate the train, validation, and test sets. The application of this technique ensures that the two groups are represented in each set in equal proportion compared to the whole dataset.

In this sense, the train, validation, and test set contain respectively 70%, 15%, and 15% of the images present in the whole dataset.

4.3 Training Schema

In this work, two methodologies of training were adopted. The first is the unsupervised training of each autoencoder model. In this regard, we first start by defining the architecture of the autoencoders.

In the second place, the three types of autoencoders were trained until their performance on the validation set does not improve for 20 epochs, which in turn caused the training to stop. The training of the autoencoders finishes the unsupervised training. Regarding this training phase, the autoencoders are trained by minimizing the mean squared error between the input and output image patches of size 512×512. For this purpose, the Adam optimization algorithm was used [42].

We also adopted a learning rate scheduler that reduced the learning rate by 90% when the validation loss has not improved in 10 epochs. This scheduler ensures that the magnitude of consecutive changes in parameter values is suited for each training phase so that local minimums of the loss function are not surpassed. Also, a checkpoint module that saves the best model based on the loss measures in the validation set and at the end of each epoch was used.

In the selection of the batch size for each type of autoencoder - given the absence or presence of attention mechanism - two aspects were considered. Firstly, as Dominic Masters and Carlo Luschi demonstrated in their work [50], for a wide range of experiments, the optimal batch size was always 32 or smaller. According to the authors, the best training stability and generalization performance were achieved at small batch sizes. In this line of thinking, we set 32 to be the maximum value for the batch size. On the other hand, the adoption of bigger batch sizes can speed up the training by decreasing computational time. That being said, we utilized batch sizes for each of the three types of autoencoders as high as possible with a ceiling of 32 and conditioned to hardware capabilities. In this sense, for the training of the standard, CBAM, and TMB autoencoders, a batch size of 16, 8, and 16, was respectively used.

For the selection of the learning rate hyperparameter, the batch size is an important variable to consider. According to the work of Granzio et al. [26], for a set of small batch sizes, the maximal learning rate should be proportional to the batch size. We consider the proportionality factor between the two variables to be $6,25 \times 10^{-5}$ which corresponds to the quotient between the default learning rate for Adam optimizer (0,001) and the batch size used for standard autoencoders (16). The carried experiments demonstrated that the adopted factor works well for the three types of autoencoders. In this sense, the adopted learning rates for the standard, CBAM, and TMB autoencoders were respectively been 0,001, 0,0005, and 0,001.

In the second stage, after the unsupervised training, the supervised training takes place. Despite using randomly cropped patches from the images associated with the training set in order to train the autoencoders, for this stage, the whole images were used. We start by obtaining the compressed representation of each image and then passed to an SVM classifier to generate the predictions.

However is important to note that the dataset with which we are working is unbalanced, being 29% and 71% of the train images to respectively belong to the healthy and cancerous target class. In this sense, class weights were computed in order to simultaneously reduce the impact of the images belonging to the majority class in the training process and penalize misclassifications made by the classifier in the minority class. The weights are calculated based on the function 4.1, being N the number of images in the training set and N_i the number of images associated with the ith target class.

$$w_i = \frac{N}{2N_i}, i = \{0, 1\} \tag{4.1}$$

It is also crucial to note that without the class weight calculation and later added to the cost function of the classifier, it would be biased to predict the most observed target class most of the time incurred in a high false positive rate.

4.4 Study Design

After detailing the training schema in the previous section we will depict how the training schema will fit in the work developed in this thesis.

After concluding the first and second stages of training, the best autoencoder architecture for each of the three types of attention is identified. This identification is based on the loss of the associated classifier measured in the validation set.

Being identified the best architectures for each type of autoencoder, the models are trained and subjected to the best set in order to retrieve some metrics of interest for future comparison.

4.5 Base Architecture of the Autoencoders

To better understand the impact of the attention modules in autoencoders, two base architectures were adopted. The base architectures of the autoencoders are composed of two distinct blocks of known architecture, which are shown in Figure 4.1.

In the mentioned figure, (a) was considered in order to build the encoder and (b) was considered to build the decoder. Both encoder and decoder are composed of a variable number of the two types of blocks. The precise number, as well as the number of filters in the convolutional layers, were considered hyperparameters that will be discussed in this chapter. Note that if the autoencoder being trained is a standard one - is not equipped with any attention module - both attention modules are skipped from the architecture of both blocks.

Concerning parametrical options, in all convolution operations, we adopted a stride and padding of 1 and a filter size of 3×3, inspired by both VGG and Xception neural networks. Max pooling layers with pool size 2×2 and stride 2 are also used in each block of the encoder in order to reduce the size of each feature map that composes the input. The selection of this pooling size is based on the fact that it is the most commonly used in pooling layers in the literature. By reducing the size of each feature map we can consider deeper inputs, allowing the encoder to capture finer features while also ensuring that the computed outputs fit in memory. The rectified linear unit or ReLU was considered as activation function to ensure that non-linear features are considered in the computation of the image representations. The ReLU activation is widely used in the deep learning community since it enjoys the property of not being bounded from above which in turn translates in the absence of gradient vanishing problems besides being computationally efficient. A batch normalization layer was also considered to make the training more stable, faster, and less susceptible to parameter initializations. This layer has become a standard component in many state-of-the-art architectures given its ability to improve training performance. The reader may also note that the batch normalization layer is inserted before the ReLU activation. Besides being a common practice in well-known deep learning models like Xception and ResNet, Ioffe and Szegedy advocate that upon the application of a non-linear activation the distribution of the data is likely to change - becoming sparse or asymmetrical - being the standardization done through the batch mean and standard deviation to probably be insufficient to eliminate the covariate shift [34].

The decoder block was built in a complementary way to the encoder. In this sense, the upsampling layers double the width and height of each feature map that composes the input by considering the nearest interpolation (default). Just like the convolutional layers in the encoder, convolution layers in the decoder have a stride and padding of one and filters of size 3×3. It is also relevant to mention that given we normalize the inputs to the unit interval before feeding them to the autoencoder, the activation associated to the last decoder block is a sigmoid function. Also in the last block of the decoder, after the sigmoid activation is applied, no attention layer is considered.

It is also of great importance to mention that skip connections were employed from every block, in the encoder, to the correspondent block in the decoder. This strategy was used to mitigate degradation problems, that arise due to the adoption of very deep models. As depicted in several publications, training very deep neural networks is a difficult task. As networks get deeper the magnitude of the gradients get smaller for initial layers which translates to very small updates for the respected weights, which in turn leads to the saturation of performance metrics and later degradation. In this line of thinking, for every encoder block, we save the output of the convolution layer and posteriorly add it (in an element-wise manner) to the output of the respected convolution, in the decoder. The first skip connection takes between the first encoder block and last decoder block, the second between the second encoder block and the penultimate decoder block, and so forth. This positioning was inspired by the work of Dong et al [20], which has shown competitive results to state-of-the-art classification architectures through the first stage of unsupervised pretraining.

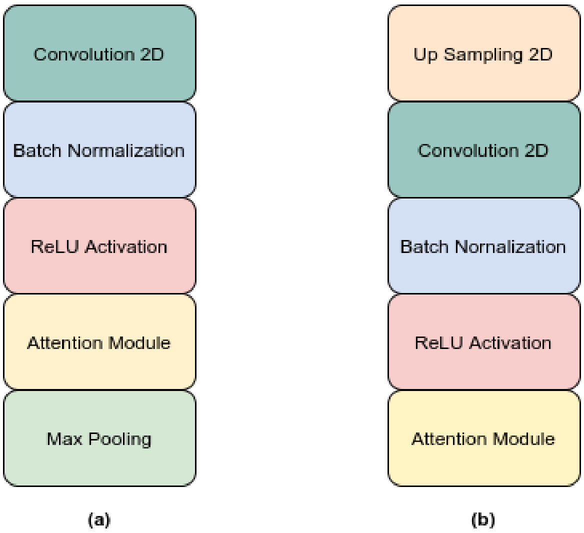

Figure 4.1: Autoencoder Blocks, being (a) and (b) associated to the encoder and decoder segments, respectively.

4.6 Choice of Classifier and Parameterization

We will start this subsection by explaining the intuition behind the way SVM classifiers operate as well as the motivation behind the adoption of this supervised learning model. Roughly put, this model aims to identify the hyperplane that best separates - in the case of binary classification - the data points associated with the two target classes while maximizing the distance between this hyperplane and its support vectors, which defines the margin of the classifier. Support vectors are the designation assigned to the data points that are closer to the hyperplane.

However, being a hyperplane described by a single linear equation it is incapable to separate non-linearly separable classes. This problem is addressed by considering a non-linear function that maps the respected inputs to a higher-dimensional space, where (hopefully) the classifier can linearly separate the 2 classes.

Through this operation mode, the SVM classifier has shown impeccable performance in a wide variety of problems, from which cancer classification is an example. In [58], Rahman et al. tested the performance of the linear SVM classifier for the binary classification of oral squamous cell carcinoma, which ended up scoring 1.0 in accuracy, precision, recall, and sensitivity metrics in the test set. Also Chu and Wang, in [14], accessed the performance of the classifier in three datasets, each associated with a different kind of cancer. Without any relevant preprocessing and being considered several feature extraction techniques, the authors reported a minimum accuracy in the test set, among the three datasets, of 0.97. The success of the SVM classifier is strongly correlated with the selection of the optimal set of hyperparameters, from which the kernel function, the C and γ parameteres occupy a central position. The kernel function is the non-linear function, above mentioned, that aims to map the inputs to a higher-dimensional space. The constant C is a regularization parameter. Lower values of C translate to a higher regularized, and therefore simpler model, which considers a larger margin. Higher values of C translate into the opposite situation. The γ values can be interpreted as the inverse of the area of influence of a single training examples. The higher the values the γ parameter takes the closer the data points need to be to the support vectors in order to get their target label.

Having been chosen the classification algorithm is critical to determine the number of features, based on which classification will be performed, given the number of samples that the train set holds. In 2004, Hua et al. published a study where for a variety of learning algorithms the error rate was calculated as a function of the sample size and size of the feature space [31]. In their study, the authors used gene-expression data of 295 patients to perform the detection of breast cancer. In the conducted experiments the authors reported the error rate surface, associated with the linear and polynomial type of SVMs, to be flatter for a wide range of sample and feature sizes compared to other classification algorithms like perceptron or the linear discriminant analysis. Furthermore, the authors identified that the minimum error rate was associated with the sample-to-feature ratios of 1.905, 1.5, and 1.(1). Given that we have 487 images for training, the mentioned ratios respectively translate to 256, 324, and 438 features. In this sense, for the task at hand, we considered 192 features (16×12), given to be the nearest number of features to the ones mentioned above.

After the determination of the number of features to use, a hyperparameter tuning process was carried out for each dataset, composed by the 487 representations associated with each training image, for a given autoencoder and through a "grid-search"method, which ensures that all parameter combinations are tested. Such optimizations were focused on the principal hyperparameters associated with an SVM classifier - the regularization parameter C and the kernel function K. When in the presence of a Gaussian kernel function, the γ parameter was also optimized, and for a polynomial kernel function, both γ and degree of the polynomial D were optimized. In this regard, all available kernel functions in the Sklearn library were tested [56]. Regarding the C and γ hyperparameters, Hsu et al. in their paper [29], found that trying exponential growing sequences of both hyperparameters led to good results, being suggested for the hyperparameter C the range $\{2^{-5}, 2^{-3}, \ldots, 2^{13}, 2^{15}\}$ and for the hyperparameter γ the range $\{2^{-15}, 2^{-13}, \ldots, 2^{1}, 2^{3}\}$. The common polynomial degree tested in the literature is 2 to 5, as such, we hypothesized that the best polynomial degree, for our task, lies in that interval [3].

4.7 Attention Modules Architecure

The CBAM module is composed of a channel attention submodule and a spatial attention submodule. Is important to note that in the channel attention submodule (a), both max and average pooling are global operations, being each feature map of the input reduced to a single number. Regarding pooling layers, it is also important to note that in the spatial attention submodule (b) both max and average pooling operations are three-dimensional. Both layers have a pool size of $1 \times 1 \times C$ where C denotes the number of feature maps that compose the input. It is also important to note that for implementation purposes the reduction ratio r is 16, the same value used in the original paper.

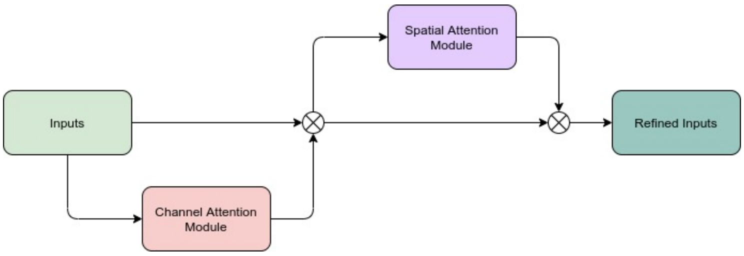

Figure 4.2: General architecture of the Convolutional Block Attention Module.

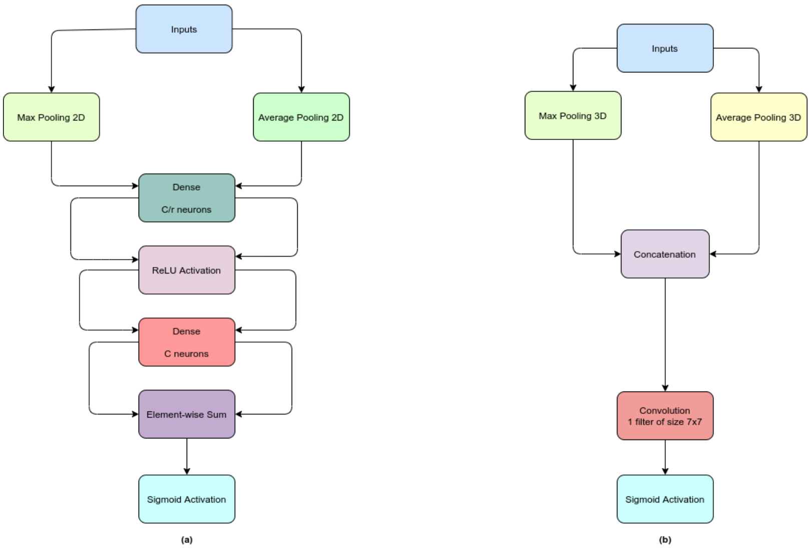

Figure 4.3: Architecture of the CBAM segments, being (a) the architecture of the channel attention submodule and (b) the architecture of the spatial attention submodule.

Followingly we move to the architecture of the TMB module.

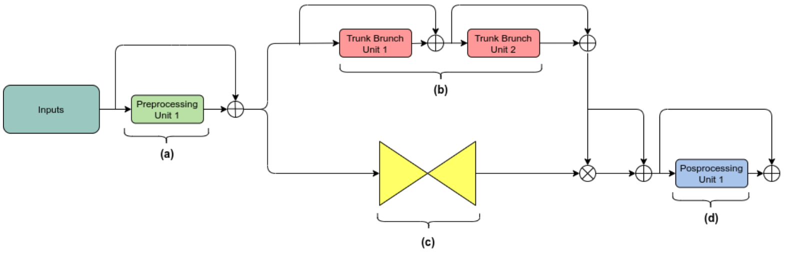

Figure 4.4: General architecture of the Trunk-Mask Branch attention, being (a), (b), (c) and (d) respectively associated to the preprocessing segment, trunk branch, mask branch and posprocessing segment.

The residual attention module is shown in Figure 4.4. As it can be seen the module is composed of three major components - the preprocessing unit, the trunk branch, and the mask branch. For implementation purposes, we considered one preprocessing unit and two units that compose the trunk branch. All residual units have the same architecture, which can be seen in Figure 4.5.

Each TMB module is associated with a given residual block, where each block has a different set of convolutional filters of size 3 × 3 and unitary padding. For each residual block, the first convolution has as many convolutional filters as the number of channels in the input to the block, while the second and third convolutions have double of the filters in the first convolution. The third convolution is only used when the dimension of the inputs differs from the dimension of its transformed. In such situations, the third convolution is applied to compute a representation of the inputs of the same dimension of its transformed, so that the element-wise sum operation between the two is defined.

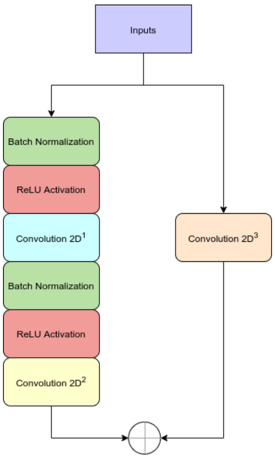

Figure 4.5: Architecture of the residual block.

Regarding the mask branch, its architecture is depicted in Figure 4.6. As can be seen below, the mask branch can be divided into two separate structures - the bottom-up and the top-down segments. The bottom-up segment aims to decrease the resolution of the inputs while extracting important features from the input through the use of residual units and the top-down segment scales the input to its original size. Note the residual units present in this branch also have the architecture shown in the Figure 4.5.

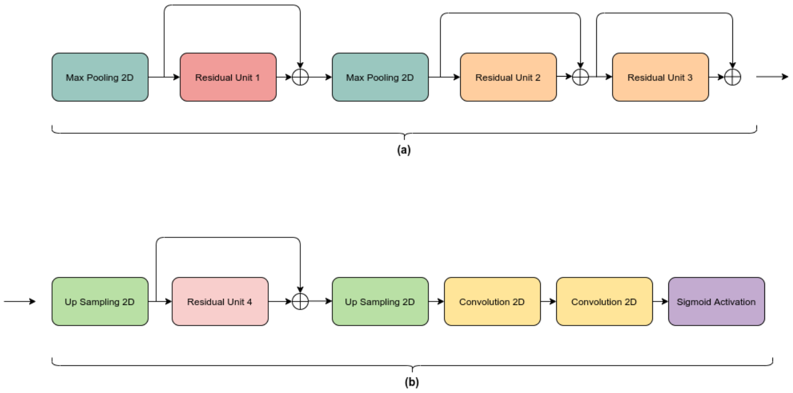

Figure 4.6: Architecture of the mask branch segment, being (a) the bottom-up segment and (b) the top-down segment architecture.

4.8 Hyperparameter Selection

Being defined the building blocks of the autoencoders as well as the details of the implementation of both attention mechanisms, we will present the hyperparameters that were considered for this study.

- **Number of blocks** The total number of blocks that compose the autoencoder. Being the encoder and decoder composed by half of this number. In the conducted experiments this hyperparameter took values in the set $\{14, 16\}$.
- **Depth of the image representation** The number of feature maps that compose the representation learned by the encoder. This hyperparameter took values in the set $\{1, 4\}$.

Being defined the number of blocks associated with each autoencoder and the number of filters present in the last convolutional layer of the encoder, which corresponds to the depth of the image representation, we will depict the logic behind such choices.

Given each block to have a max-pooling layer with pool size 2×2, the number of blocks has a direct impact on the size of each feature map that composes the encoding. The reader needs to recall that we have established the number of features of the classification problem to be 192, which in turn translates to an autoencoder with a minimum of 14 blocks, being the representation composed by a single feature map $(16 \times 12 \times 1)$. In order to adopt other autoencoder architectures, we also adopted an autoencoder composed of 16 blocks, being in this situation the representation is composed of four feature maps (8×6×4). An autoencoder architecture with 18 blocks was considered but due to being the train inputs of size 512×512, at some point, the intermediate output would be impractical to be processed by the bottom-up segment in the mask branch of the TMB attention module, given it small shape. In this manner, both the number of blocks and the depth of the encodings for each one are established.

Based on the two hyperparameters we considered 12 autoencoders architectures from which one half will be trained on grayscale image patches and the other half in colored RGB image patches. Each autoencoder is associated with a given triplet-type of image patch (either grayscale or RGB) where it will be trained, the number of blocks, and the depth of the respected representation.

Being defined the number of blocks of the encoder and the depth of the image representation learned by the encoder, we need to set the number of filters associated with each convolutional layer in each block. To do so, two different methodologies were considered depending on the number of blocks of the encoder and the number of filters of the last convolution in the encoder.

The selection of the number of filters is based on the principle that if the number of filters associated with the kth block of the encoder is n_k then the number of filters associated with the $(k+1)$th block must be at least $2n_k$. Nevertheless, in some architectures, the number of blocks is too high for the number of convolutional filters associated with the last block of the encoder, for the property to hold. In such situations we opt for dividing the encoder into two separate segments - the first segment is characterized by holding that property while the second segment is characterized by the complementary property. Such property states that if the number of filters associated with the kth block of the encoder is n_k then the $(k+1)$th block has $\frac{1}{2}n_k$ filters associated. The first segment aims to compute from low-level to high-level features while the second segment aims to distill that knowledge in progressively shallower outputs. This choice of architecture for the autoencoders was inspired by the work of Kucharski et al. [45].

In that work, the authors proposed to perform the segmentation of dermatopathological images based on two convolutional autoencoders. Being one of the autoencoders optimized to reconstruct such images while simultaneously learning useful image representations that can ease the segmentation process. Such pipeline achieved state-of-the-art performances in the segmentation task. In particular, the architecture of the second segment of our autoencoders - in which each block reduces the dimension of the input 8 times - is based on the third block of their autoencoder which also performs the same reduction.

In this sense, we hypothesize that if their autoencoder could learn useful representations for segmentation maybe a similar architecture can also learn useful representations for classification.

Instance id	**Number of Blocks**	**Code Depth**	**Filter Set**	**Size Reduction**
Gray_14_1	14	1	8, 16, 32, 16, 8, 4, 1	16384
Gray_16_4	16	4	8, 16, 32, 64, 32, 16, 8, 4	16384

Table 4.1: Autoencoder Architecture for Grayscale images. The Instance id column is built with the goal for the reader to identify a specific autoencoder model - the first part identifies in which type of data the autoencoder was trained on, the second is the number of blocks, and the third the depth of the representation produced by the autoencoder.

Instance id	**Number of Blocks**	**Image Representation Depth**	**Filter Set**	**Size Reduction**
Rgb_14_1	14	1	8, 16, 32, 16, 8, 4, 1	49152
Rgb_16_4	16	4	8, 16, 32, 64, 32, 16, 8, 4	49152

Table 4.2: Autoencoder Architecture for Rgb images. The Instance id column is built with the goal for the reader to identify a specific autoencoder model - the first part identifies in which type of the data the autoencoder was trained on, the second is the number of blocks and the third the depth of the representation produce by the autoencoder.

Chapter 5

Analysis of Results

In this chapter, as the name suggests, we will proceed with the analysis of results associated with the conducted experiences, which were depicted in the previous chapter. In this sense, we have chosen to split this chapter into two subsections - being the first associated with the train, validation, and testing in grayscale images and the second in colored RGB images.

In a second stage, each subsection is composed of two subsubsections. Being the first dedicated to the analysis of the outputs of the autoencoders during the unsupervised training and the second is dedicated to the analysis of the outputs associated with the supervised training.

5.1 Grayscale Analysis

We will begin this section by displaying, for each autoencoder and autoencoder type - Standard, CBAM, and TMB - the best validation loss and the associated train loss. All analyses depicted in this section - train, validation, and test - are based on single-channel grayscale images.

	Standard		**CBAM**		**TMB**	
Instance id	**Train**	**Validation**	**Train**	**Validation**	**Train**	**Validation**
Gray_14_1	0,0014	0,000254	0,0021	0,000246	0,0075	0,000618
Gray_16_4	0,0016	0,000205	0,0014	0,000230	0,0071	0,000736

Table 5.1: Training and validation losses based on grayscale images. Note that the loss used for training and validation purposes is the mean squared error.

At a first glance, we may suspect that the TMB autoencoders perform worst than the standard and CBAM autoencoders, both in train and validation sets. In order to confirm (or refute) this hypothesis we performed a Kruskal-Wallis statistical test [44]. This test aims to calculate the level of agreement between the null hypothesis - which states that all samples have been drawn from the same distribution - and the observed data. By performing such test in the train and validation sets, the p-value is respectively 0,165 and 0,180. This proves that there is no statistical difference between the three autoencoders in the train and validation sets. This in turn translates to the fact that attention-armed autoencoders do not perform better then the standard ones in the reconstruction task.

5.1.1 Input/Output Comparison

In this subsection, we aim to compare the input images fed to the autoencoders with their respected reconstruction. This comparison also aims to visually inspect the differences in the image quality associated with the different autoencoders as well as get a sense of whether the autoencoders are well trained.

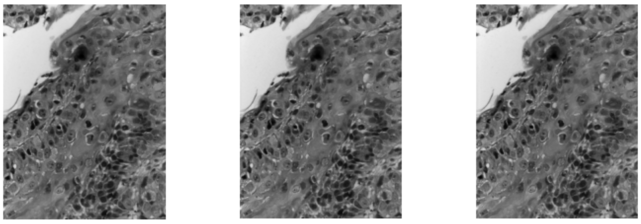

Figure 5.1: Comparison between the input image and the output images associated to the 2 standard autoencoders. The left image is the original one fed as input to the autoencoders. The central and right images are the outputs of the standard-0 and standard-1 autoencoders.

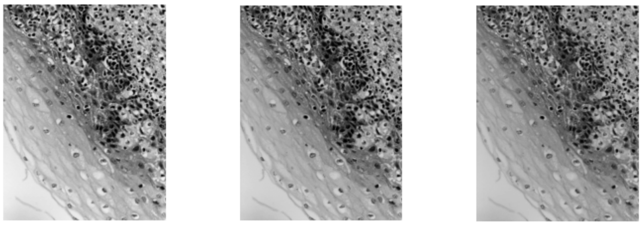

Figure 5.2: Comparison between the input image and the output images associated to the 2 CBAM autoencoders. The left image is the original one fed as input to the autoencoders. The central and right images are the outputs of the CBAM-0 and CBAM-1 autoencoders.

When examining the three visualizations we can see that the autoencoders were perfectly capable of reconstructing the input images, which in turn translates to the fact that all autoencoders are well trained.

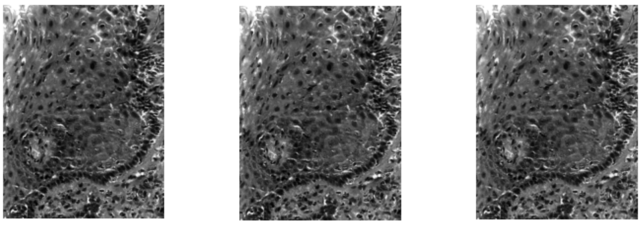

Figure 5.3: Comparison between the input image and the output images associated to the 2 TMB autoencoders. The left image is the original one fed as input to the autoencoders. The central and right images are the outputs of the TMB-0 and TMB-1 autoencoders.

5.1.2 Manifold Analysis

After the display and analysis of the respected mappings between input images and output reconstructions, we will move our analysis to the manifold learned by each autoencoder from the train set. Just like in the previous subsection, the images used in order to visualize the learned manifold for each autoencoder, are from the test set.

Regarding the visualization pipeline, it is composed of three steps. We begin by standardizing the features, so that each follows a standard normal distribution, using their mean and standard deviation to do so. Given the dataset is composed of 192 features, we considered the use of a preliminary dimension technique to reduce the number of features to a reasonable number while suppressing some noise. With this intent, we opted for Principal Component Analysis [51] with a hyperparameter of 50 components, as suggested by Scikit-learn [56]. Lastly, we used t-SNE [48] to map the n-dimensional dataset to a 2-dimensional one. In this matter, we considered both random initialization (the default) and PCA initialization of the embedding, which is theoretically associated with a more stable learning process. The reason behind the selection of this dimensionality reduction technique lies in the fact of it is a nonlinear method aiming to maintain neighborhood data points and local structure. On the analysis of the manifold learned by the autoencoders, we are interested in seeing if images belonging to the same target class are near each other, being the global structure of the manifold of not significant importance. To identify the best pipeline for visualization we considered a metric to optimize, being the silhouette score calculated among all points. In this way, the best visualization pipeline for a given autoencoder is the one that maximizes the silhouette score among the others.

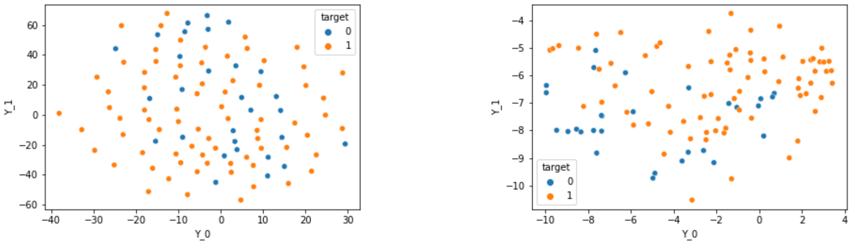

Figure 5.4: Standard autoencoders manifolds associated to grayscale images. The left and right plots are respectively associated to the standard-0 and standard-1 autoencoders. Blue data points are associated to healthy images and orange to cancerous ones.

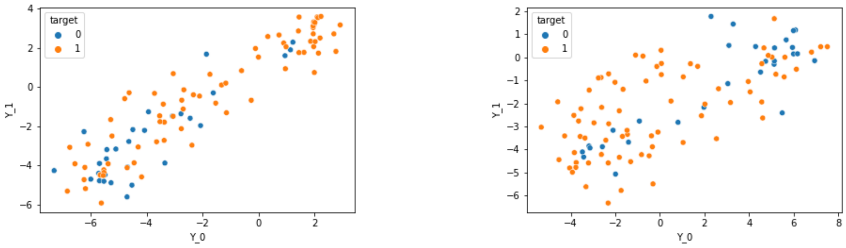

Figure 5.5: CBAM autoencoders manifolds associated to grayscale images. The left and right plots are respectively associated to the CBAM-0 and CBAM-1 autoencoders. Blue data points are associated to healthy images and orange to cancerous ones.

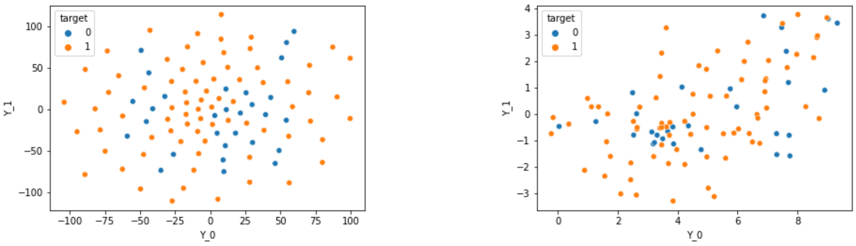

Figure 5.6: TMB autoencoders manifold associated to grayscale images. The left and right plors are respectively associated to the TMB-0 and TMB-1 autoencoders. Blue data points are associated to healthy images and orange to cancerous ones.

As the reader can see from the above visualizations, both autoencoders associated to the CBAM attention module were able to learn a interesting manifold of the data points, since they are reasonably well separated according to the classes of the target. On the other hand, despite the points associated to the healthy and cancerous to be perfectly mixed in the visualizations associated to the standard and TMB autoencoders, we cannot claim that the manifolds learned by the mentioned autoencoders are not meaningful. The mentioned visualizations of the manifolds may have been distorted due to the application of the visualization techniques.

5.1.3 Classification Analysis

While in the previous two subsections we analyzed the outputs of the trained autoencoders, in this subsection we will analyze the outputs of the SVM classifiers. In the first stage, we will identify the autoencoder that is associated with the best validation f1-score. The motivation behind the choice of this metric is centered around the fact of the datasets at hand are imbalanced. In a second stage, we will analyze the predictions of the three SVMs through a set of four metrics - positive predictive value (PPV), negative predictive value (NPV), sensitivity, and specificity.

As mentioned in the previous chapter and to identify the three best autoencoders, several hyperparameter tuning was conducted. In the below table we show, for each autoencoder, the optimal hyperparameter set as well as the associated train and validation f1 metrics.

Instance id	**Attention**	C_{opt}	K_{opt}	D_{opt}	γ_{opt}	**Train**	**Validation**
Gray_14_1	Standard	0,03125	Polynomial	3	0,125	0,8615	0,8488
Gray_16_4	Standard	0,03125	Polynomial	4	0,5	0,9583	0,8701
Gray_14_1	CBAM	8	Gaussian	-	0,5	0,9898	0,8193
Gray_16_4	CBAM	2	Gaussian	-	0,125	0,9394	0,8435
Gray_14_1	TMB	2	Gaussian	-	8	0,9898	0,8182
Gray_16_4	TMB	8	Gaussian	-	8	0,9898	0,8050

Table 5.2: Best SVM hyperparameters, for each autoencoder, and the associated train and validation f1-scores based on grayscale images. C is the regularization paramter while K identifies the kernel function. When the last is polynomial, D is its degree. γ is the gamma paramter associated to the gaussian and polynomial kernels. The Attention column identifies the attention module associated with a given autoencoder.

From the analysis of the above table, we can see that the standard autoencoders achieved better performance, regarding the f1-score metric, in the validation set compared with the other autoencoders. It is also important to declare that for the subsequent analysis we will only consider the best autoencoder from each type of attention module. We will now pass on the description of the four metrics used in the evaluation of the classification predictions.

The positive predictive value, also called PPV, is the proportion of correctly classified cancerous images among the images that were classified as cancerous. This metric aims to approximate the probability of an image being cancerous given that the model predicted that it is cancerous. The negative predictive value, or NPV, implements the same logic associated with the predictive value but regarding the healthy images. In this sense, it approximates the probability of a given image to be healthy given that the model classified it as healthy.

The positive and negative predictive values aim to approximate the probability of an image to belong to a given class, conditioned on the prediction of the model. Sensitivity and specificity aim to approximate the probability of the image to belong to a class but are conditioned on the real classification of the image or ground truth.

The sensitivity aims to approximate the probability of a given image to be classified as cancerous given that it is effectively cancerous. The specificity applies the same logic of sensitivity to healthy images. That is, approximates the probability of a given image te classified as healthy given that it is healthy.

Instance id	Attention	PPV	NPV	Specificity	Sensitivity
Gray_16_4	Standard	0,827	0,652	0,517	0,893
Gray_16_4	CBAM	0,843	0,529	0,621	0,787
Gray_14_1	TMB	0,718	0,0	0,0	0,987

Table 5.3: Classification metrics associated with the best performance autoencoders in grayscale images.

From the above table, we can begin by noting that different types of autoencoders are better at the recognition of a given target class than others. For instance, the representation learned by the standard autoencoder led to the best NPV metric which translates into higher confidence of a given image to be healthy if the SVM classifier, fed on the features produced by a standard autoencoder, predicts so. On the other hand, the CBAM autoencoder is associated with the highest metrics of PPV and specificity. This autoencoder being the best concerning the specificity metric proves that the features learned by the CBAM autoencoder can correctly identify the majority of the healthy images. On the other hand, this autoencoder being associated with the best PPV metric demonstrates the confidence of a positive prediction from an SVM classifier based on a representation of an image through a CBAM autoencoder. Lastly, the TMB autoencoder is associated with the best sensitivity metrics, in this manner, the classifier associated with that autoencoder was able to identify 98,7% of the cancerous images. Neverhteless, the classifier trained under the representations learned by the TMB autoencoder scored 0,0 on the NPV and specificity metrics. This is explained by the fact that the classifier classified all test images as positive but one, which is a false negative observation. In this sense, the number of true negative images is zero which in turn translates to the mentioned metrics to be zero.

We end this analysis by stating that the best feature extractor method depends on the performance metric that we wish to optimize. Nevertheless and globally, the CBAM autoencoder seems to be the better performing since outperforms the other two autoencoders in 2 out of the 4 considered metrics.

5.2 Color Analysis

This section was built similarly to the previous one, starting our analysis in the outputs of the autoencoders and ending it in the analysis of the outputs of the classifiers. As the name suggests the previous section was focused on grayscale images with 1 channel while this section focus on RGB images with 3 channels.

In this manner, we will start by displaying, for each autoencoder and autoencoder type - standard, CBAM, and TMB - the best validation and the associated train losses. While in the previous section all analyses were based on grayscale images, in this chapter they are associated with three-channel RGB images.

	Standard		**CBAM**		**TMB**	
Instance id	**Train**	**Validation**	**Train**	**Validation**	**Train**	**Validation**
Rgb_14_1	0,0014	0,000591	0,0020	0,000540	0,0021	0,000826
Rgb_16_4	0,0028	0,001681	0,0023	0,000498	0,0017	0,000738

Table 5.4: Training and validation mean squared error losses for each autoencoder type and instance based on RGB images.

From the above table, we can start by noting that both CBAM autoencoders outperformed the other 4 models in the validation set, being the ones that better generalize across all other trained autoencoders. It is also worth noting that the performance of the TMB autoencoders greatly improved when considering colored 3-channel images as input compared with grayscale images.

5.2.1 Input/Output Comparison

In this subsection, we will compare the input images to the reconstruction outputs for each trained autoencoder. As mentioned in the previous section, the visual inspection of the input and output images can identify potential problems in the train and also if the autoencoders are well trained.

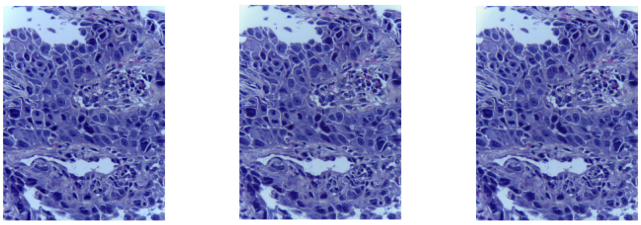

Figure 5.7: Comparison between the input image and the output images associated to the 2 standard autoencoders. The left image is the original one fed as input to the autoencoders. The central and right images are associated to the standard-0 and standard-1 autoencoders.

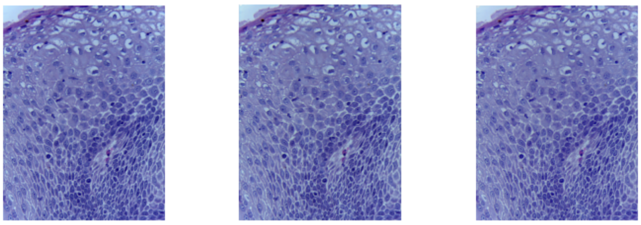

Figure 5.8: Comparison between the input image and the output images associated to the 2 CBAM autoencoders. The left image is the original one fed as input to the autoencoders. The central and right images are associated to the CBAM-0 and CBAM-1 autoencoders.

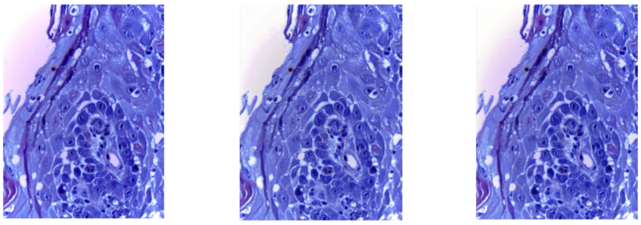

Figure 5.9: Comparison between the input image and the output images associated to the 2 TMB autoencoders. The left image is the original one fed as input to the autoencoders. The central and right images are associated to the TMB-0 and TMB-1 autoencoders.

Just like the autoencoders trained in grayscale images, the comparison between the input and output images reveals that all autoencoders are well trained.

5.2.2 Manifold Analysis

Finishing our analysis on the input and output comparison for the selected pairs of autoencoders, we will conduct a visual analysis on the learned manifold for the selected pairs of autoencoders. To produce the below visualizations a visualization pipeline was considered, which is explained in the previous section in the subsection of the same name as this one.

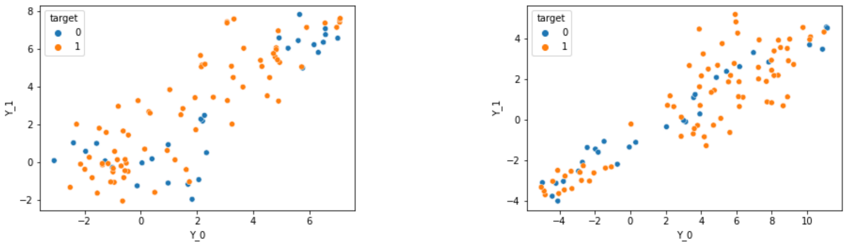

Figure 5.10: Standard autoencoders manifolds associated to RGB images. The left and right plots are respectively associated to the standard-0 and standard-1 autoencoders. Blue data points are associated to healthy images and orange to cancerous images.

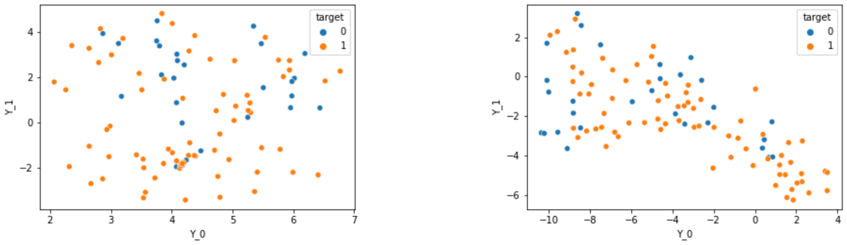

Figure 5.11: CBAM autoencoders manifolds associated to RGB images. The left and right plots are respectively associated to the CBAM-0 and CBAM-1 autoencoders. Blue data points are associated to healthy images and orange to cancerous images.

We can begin the analysis of the above plots by seeing that both visualizations associated to the TMB autoencoders are messy, being the points of both classes poorly clustered. This situation can either be because the TMB autoencoders were uncapable of learning a meaningful representation of the data or because PCA and t-SNE of capturing the structure of the manifolds. The same situation does not hold for the standard and CBAM types of autoencoders. It is interesting to see that in both visualizations associated with standard autoencoders, the data points associated with the healthy images seem to be divided into two clusters while the data points associated with the cancerous image

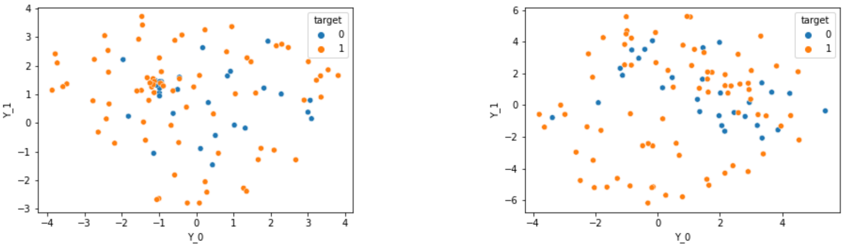

Figure 5.12: TMB autoencoders manifold associated to RGB images. The left and right plors are respectively associated to the TMB-0 and TMB-1 autoencoders. Blue data points are associated to healthy images and orange to cancerous images.

seem to define a single and longer cluster. The manifold learned by the CBAM-0 is of no particular utility since all the data points seem to be mixed up, while in the manifold associated with the CBAM-1 is of greater interest. In the last, the majority of the healthy data points seem to surround the data points associated with the cancerous images.

5.2.3 Classification Analysis

Similar to the previous section, where we discuss the metrics associated with the grayscale images, in this subsection, we focus on the discussion of the classification metrics based on RGB images. For that purpose, we begin with the determination of the SVM hyperparameters that are associated with the highest f1-score, measured on the validation set, through hyperparameter tuning. After doing so, we identify the autoencoder - from each attention type - associated with the best performance. Lastly, we analyze the performance of the three SVM classifiers through four metrics, when trained in the representations associated with the three autoencoders.

In the below table we display, for each autoencoder architecture and attention type, the hyperparameter set that led to the highest validation f1-score.

Analyzing the above table we can see that, as opposed to the train in grayscale images, the CBAM autoencoders slightly outperform the standard autoencoders. Again, the f1-scores associated with the TMB autoencoders are the worst compared with the other two groups. We will now focus our analysis on the predictions of the SVM classifier that better performed in the validation set, for each attention module.

From a first glance at the table, we can see that the TMB autoencoder is underperformed by the standard and CBAM types of autoencoders. As mentioned before, we consider a possible explanation for this behavior the fact that the trunk-mask branch attention mechanism was designed to incorporate feed-forward convolutional neural network architectures in image classification tasks. On the other hand, we can see that the CBAM autoencoders outperform the standard autoencoder in NPV and sensitivity

Instance id	**Attention**	C_{opt}	K_{opt}	D_{opt}	γ_{opt}	**Train**	**Validation**
Rgb_14_1	Standard	0,5	Gaussian	-	0,125	0,8973	0,8662
Rgb_16_4	Standard	0,5	Polynomial	3	0,03125	0,86	0,8506
Rgb_14_1	CBAM	32	Polynomial	5	8	0,8506	0,8571
Rgb_16_4	CBAM	0,125	Polynomial	2	0,125	0,8668	0,8727
Rgb_14_1	TMB	0,5	Gaussian	-	8	0,9433	0,8208
Rgb_16_4	TMB	2	Polynomial	4	0,5	0,8549	0,8263

Table 5.5: Best SVM hyperparameters, for each autoencoder, and the associated train and validation f1-scores based on RGB images. C is the regularization parameter while K identifies the kernel function. When the kernel function is a polynomial, D is its degree. The Attention column identifies the attention module associated with a given autoencoder.

Instance id	**Attention**	**PPV**	**NPV**	**Specificity**	**Sensitivity**
Rgb_14_1	Standard	0,838	0,667	0,552	0,893
Rgb_16_4	CBAM	0,791	0,769	0,345	0,960
Rgb_16_4	TMB	0,709	0,222	0,138	0,813

Table 5.6: Classification metrics associated with the best performance autoencoders in RGB images.

metrics while the standard autoencoder outperforms the last in the PPV and specificity metrics. In this way, the two autoencoders complement each other. The SVM classifier when trained in the compressed representations learned by a standard autoencoder can identify the higher proportion of the healthy images, which corresponds to more than half of the total amount of healthy images in the test set. Concerning the PPV, 83,8% of the images classified as cancerous are cancerous, this translates to high confidence in the cancerous status of an image if the classifier makes such prediction. The opposite situation takes place regarding the CBAM autoencoder. The SVM classifier trained on its embedding was able to identify 96% of the cancerous images but being the negative predictions associated with a higher level of confidence.

We finish this chapter by declaring that the best autoencoder method depends on the performance metric that we wish to optimize. If we focus to reduce the number of false positive samples, the standard autoencoder is a better option given that it is associated with higher confidence in the positive images. If on the other hand, we focus on the reduction of false negative samples, the CBAM autoencoder is a better option.

Chapter 6

Conclusions and Future Work

Throughout this thesis, several conclusions were reached concerning the use of attention mechanisms associated with autoencoder models for the extraction of features.

The first is that no improvements were observed, regarding the quality of the reconstructed images, if the feature extraction process was carried by autoencoders armed with attention mechanisms by comparison with standard autoencoders. This conclusion is supported by the fact that there is no statistical significance that the reconstructive train errors, associated with the 3 types of autoencoders for the 2 types of data, come from more than one distribution. Despite achieving similar performances regarding the reconstruction of images, by plotting the manifolds of each autoencoder, it is possible to see that three types of autoencoders learned different representations of the data. For future work purposes, it would be interesting to study the properties of the learned manifolds through the application of different clustering algorithms. This study would give some insight into the physical properties of the tissue considered for each type of autoencoder.

The conclusion that the autoencoders learned different aspects of the data is, besides the plotting of the manifolds, also supported by the classification metrics associated with the 3 different best autoencoders. Based on RGB images, the best standard autoencoder outperforms the other 2 in the PPV and specificity metrics while the best CBAM autoencoder outperforms the others in NPV and sensitivity metrics. On the other hand, based on grayscale images, the opposite takes place. Despite the observation of this pattern in the calculated metrics, due to the number of conducted experiences, we cannot conclude that standard and CBAM autoencoders complement each other in the classification. The extent by which the standard and CBAM autoencoders complement each other in classifications is, at this point, a candidate for future work. It is also a reason for our questioning the fact that the TMB autoencoders, which are associated with the highest number of parameters, are associated with the worst classification metrics. Another candidate for future work would be an analysis of TMB autoencoders to understand the reason for such classification performances.

We finish this section by saying that the best autoencoder, for classification purposes and conditioned to the executed experiences, depends on the metrics that we wish to optimize. If we are classifying colored images, the standard autoencoders learn features that allow the identification of a greater proportion of cancerous images while the same situation applies to the CBAM autoencoders and healthy images.

Bibliography

[1] P. Afshar, A. Mohammadi, K. N. Plataniotis, A. Oikonomou, and H. Benali. "From Handcrafted to Deep-Learning-Based Cancer Radiomics: Challenges and Opportunities." In: *IEEE Signal Processing Magazine* 36.4 (2019), pp. 132–160. doi: 10.1109/MSP.2019.2900993.

[2] A. Aggarwal, N. Das, and I. Sreedevi. "Attention-guided deep convolutional neural networks for skin cancer classification." In: Nov. 2019, pp. 1–6. doi: 10.1109/IPTA.2019.8936100.

[3] S. Ali and K. Smith-Miles. "On optimal degree selection for polynomial kernel with support vector machines: Theoretical and empirical investigations." In: *KES Journal* 11 (Feb. 2007), pp. 1–18. doi: 10.3233/KES-2007-11101.

[4] M. Anthimopoulos, S. Christodoulidis, L. Ebner, A. Christe, and S. Mougiakakou. "Lung Pattern Classification for Interstitial Lung Diseases Using a Deep Convolutional Neural Network." In: *IEEE Transactions on Medical Imaging* 35 (Feb. 2016), pp. 1–1. doi: 10.1109/TMI.2016.2535865.

[5] A. Araujo, W. D. Norris, and J. Sim. "Computing Receptive Fields of Convolutional Neural Networks." In: *Distill* (2019).

[6] U. Avni, H. Greenspan, E. Konen, M. Sharon, and J. Goldberger. "X-ray Categorization and Retrieval on the Organ and Pathology Level, Using Patch-Based Visual Words." In: *IEEE transactions on medical imaging* 30 (Nov. 2010), pp. 733–46. doi: 10.1109/TMI.2010.2095026.

[7] E. Ayan and H. M. Ünver. "Data augmentation importance for classification of skin lesions via deep learning." In: *2018 Electric Electronics, Computer Science, Biomedical Engineerings' Meeting (EBBT)*. 2018, pp. 1–4. doi: 10.1109/EBBT.2018.8391469.

[8] D. Bahdanau, K. Cho, and Y. Bengio. "Neural Machine Translation by Jointly Learning to Align and Translate." In: *ArXiv* 1409 (Sept. 2014).

[9] D. Ballard. "Modular Learning in Neural Networks." In: *AAAI*. 1987.

[10] A. Bentaieb and G. Hamarneh. "Predicting Cancer with a Recurrent Visual Attention Model for Histopathology Images: 21st International Conference, Granada, Spain, September 16–20, 2018, Proceedings, Part II." In: Sept. 2018, pp. 129–137. isbn: 978-3-030-00933-5. doi: 10.1007/978-3-030-00934-2_15.

[11] L. Chen, H. Zhang, J. Xiao, L. Nie, J. Shao, W. Liu, and T.-S. Chua. “SCA-CNN: Spatial and Channel-Wise Attention in Convolutional Networks for Image Captioning.” In: July 2017, pp. 6298–6306. DOI: 10.1109/CVPR.2017.667.

[12] M. Chen, X. Shi, Y. Zhang, D. Wu, and M. Guizani. “Deep Features Learning for Medical Image Analysis with Convolutional Autoencoder Neural Network.” In: *IEEE Transactions on Big Data* PP (June 2017), pp. 1–1. DOI: 10.1109/TBDATA.2017.2717439.

[13] Q. Chen, X. Xu, S. Hu, X. Li, Q. Zou, and Y. Li. “A transfer learning approach for classification of clinical significant prostate cancers from mpMRI scans.” In: Mar. 2017, 101344F. DOI: 10.1117/12.2279021.

[14] F. Chu and L. Wang. “Application of Support Vector Machine to Cancer Classification with Microarray Data.” In: *International journal of neural systems* 15 (Jan. 2006), pp. 475–84. DOI: 10.1142/S0129065705000396.

[15] E. Cosatto, M. Miller, H. Graf, and J. Meyer. “Grading Nuclear Pleomorphism on Histological Micrographs.” In: vol. 7. Jan. 2009, pp. 1–4. DOI: 10.1109/ICPR.2008.4761112.

[16] N. D.B. “Cancer Cell Nucleus: An Insight.” In: *Journal of Molecular Biomarkers Diagnosis* 8 (Jan. 2017). DOI: 10.4172/2155-9929.S2-026.

[17] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. “Imagenet: A large-scale hierarchical image database.” In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee. 2009, pp. 248–255.

[18] O. Deperlioglu, U. Köse, D. Gupta, A. Khanna, and A. Kumar. “Diagnosis of heart diseases by a secure Internet of Health Things system based on Autoencoder Deep Neural Network.” In: *Computer Communications* 162 (Aug. 2020). DOI: 10.1016/j.comcom.2020.08.011.

[19] A. Dobrenkii, R. Kuleev, A. Khan, A. Ramirez Rivera, and A. M. Khattak. “Large residual multiple view 3D CNN for false positive reduction in pulmonary nodule detection.” In: *2017 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*. 2017, pp. 1–6. DOI: 10.1109/CIBCB.2017.8058549.

[20] L.-F. Dong, Y.-Z. Gan, X.-L. Mao, Y.-B. Yang, and C. Shen. “Learning Deep Representations Using Convolutional Auto-Encoders with Symmetric Skip Connections.” In: Apr. 2018, pp. 3006–3010. DOI: 10.1109/ICASSP.2018.8462085.

[21] S. Ebiaredoh-Mienye, E. Esenogho, and T. Swart. “Integrating Enhanced Sparse Autoencoder-Based Artificial Neural Network Technique and Softmax Regression for Medical Diagnosis.” In: *Electronics* 9 (Nov. 2020). DOI: 10.3390/electronics9111963.

[22] M. Frid-Adar, I. Diamant, E. Klang, M. Amitai, J. Goldberger, and H. Greenspan. "GAN-based synthetic medical image augmentation for increased CNN performance in liver lesion classification." In: *Neurocomputing* 321 (2018), pp. 321–331. ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2018.09.013. URL: https://www.sciencedirect.com/science/article/pii/S0925231218310749.

[23] R. J. Gillies, P. E. Kinahan, and H. Hricak. "Radiomics: Images Are More than Pictures, They Are Data." In: *Radiology* 278.2 (2016). PMID: 26579733, pp. 563–577. DOI: 10.1148/radiol.2015151169. eprint: https://doi.org/10.1148/radiol.2015151169. URL: https://doi.org/10.1148/radiol.2015151169.

[24] L. Gong, S. Jiang, Z. Yang, G. Zhang, and L. Wang. "Automated pulmonary nodule detection in CT images using 3D deep squeeze-and-excitation networks." In: *International Journal of Computer Assisted Radiology and Surgery* 14 (Apr. 2019), pp. 1–11. DOI: 10.1007/s11548-019-01979-1.

[25] M. Gour, S. Jain, and T. Kumar. "Residual learning based CNN for breast cancer histopathological image classification." In: *International Journal of Imaging Systems and Technology* (Jan. 2020). DOI: 10.1002/ima.22403.

[26] D. Granziol, S. Zohren, and S. Roberts. "Learning rates as a function of batch size: A random matrix theory approach to neural network training." In: *arXiv preprint arXiv:2006.09092* (2020).

[27] X. Guo, X. Liu, E. Zhu, and J. Yin. "Deep Clustering with Convolutional Autoencoders." In: Oct. 2017, pp. 373–382. ISBN: 978-3-319-70095-3. DOI: 10.1007/978-3-319-70096-0_39.

[28] K. He, X. Zhang, S. Ren, and J. Sun. "Deep residual learning for image recognition." In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[29] C.-w. Hsu, C.-c. Chang, and C.-J. Lin. "A Practical Guide to Support Vector Classification Chih-Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin." In: (Nov. 2003).

[30] J. Hu, L. Shen, and G. Sun. "Squeeze-and-Excitation Networks." In: June 2018, pp. 7132–7141. DOI: 10.1109/CVPR.2018.00745.

[31] J. Hua, Z. Xiong, J. Lowey, E. Suh, and E. R. Dougherty. "Optimal number of features as a function of sample size for various classification rules." In: *Bioinformatics* 21.8 (Nov. 2004), pp. 1509–1515. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/bti171. eprint: https://academic.oup.com/bioinformatics/article-pdf/21/8/1509/691983/bti171.pdf. URL: https://doi.org/10.1093/bioinformatics/bti171.

[32] P.-W. Huang and Y. Lai. "Effective segmentation and classification for HCC biopsy images." In: *Pattern Recognition* 43 (Apr. 2010), pp. 1550–1563. DOI: 10.1016/j.patcog.2009.10.014.

[33] D. Hubel and T. Wisel. "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex." In: *The Journal of physiology* 160 (Jan. 1962), pp. 106–154. DOI: 10.1113/jphysiol.1962.sp006837.

[34] S. Ioffe and C. Szegedy. "Batch normalization: Accelerating deep network training by reducing internal covariate shift." In: *International conference on machine learning*. PMLR. 2015, pp. 448–456.

[35] Q. Jin, Z.-P. Meng, C. Sun, L. Wei, and R. Su. "RA-UNet: A Hybrid Deep Attention-Aware Network to Extract Liver and Tumor in CT Scans." In: *Frontiers in Bioengineering and Biotechnology* 8 (2020).

[36] V. Kadam and S. Jadhav. "Feature Ensemble Learning Based on Sparse Autoencoders for Diagnosis of Parkinson's Disease: Proceedings of ICCASP 2018." In: Jan. 2019, pp. 567–581. ISBN: 978-981-13-1512-1. DOI: 10.1007/978-981-13-1513-8_58.

[37] V. Kadam, S. Jadhav, and K. Vijayakumar. "Breast Cancer Diagnosis Using Feature Ensemble Learning Based on Stacked Sparse Autoencoders and Softmax Regression." In: *Journal of Medical Systems* 43 (July 2019). DOI: 10.1007/s10916-019-1397-z.

[38] M. Kallenberg, K. Petersen, M. Nielsen, A. Ng, P. Diao, C. Igel, C. Vachon, K. Holland, N. Karssemeijer, and M. Lillholm. "Unsupervised Deep Learning Applied to Breast Density Segmentation and Mammographic Risk Scoring." In: *IEEE Transactions on Medical Imaging* 35 (Feb. 2016), pp. 1–1. DOI: 10.1109/TMI.2016.2532122.

[39] X. Kang, X. Liu, X. Nie, X. Xi, and Y. Yin. "Attention Model Enhanced Network for Classification of Breast Cancer Image." In: *arXiv e-prints* (Oct. 2020). URL: https://ui.adsabs.harvard.edu/abs/2020arXiv201003271K.

[40] D. S. Kermany, M. Goldbaum, W. Cai, C. C. Valentim, H. Liang, S. L. Baxter, A. McKeown, G. Yang, X. Wu, F. Yan, J. Dong, M. K. Prasadha, J. Pei, M. Y. Ting, J. Zhu, C. Li, S. Hewett, J. Dong, I. Ziyar, A. Shi, R. Zhang, L. Zheng, R. Hou, W. Shi, X. Fu, Y. Duan, V. A. Huu, C. Wen, E. D. Zhang, C. L. Zhang, O. Li, X. Wang, M. A. Singer, X. Sun, J. Xu, A. Tafreshi, M. A. Lewis, H. Xia, and K. Zhang. "Identifying Medical Diagnoses and Treatable Diseases by Image-Based Deep Learning." In: *Cell* 172.5 (2018), 1122–1131.e9. ISSN: 0092-8674. DOI: https://doi.org/10.1016/j.cell.2018.02.010. URL: https://www.sciencedirect.com/science/article/pii/S0092867418301545.

[41] P. Khatamino, I. Canturk, and L. Ozyilmaz. "A Deep Learning-CNN Based System for Medical Diagnosis: An Application on Parkinson's Disease Handwriting Drawings." In: Oct. 2018, pp. 1–6. DOI: 10.1109/CEIT.2018.8751879.

[42] D. P. Kingma and J. Ba. "Adam: A method for stochastic optimization." In: *arXiv preprint arXiv:1412.6980* (2014).

[43] A. Krizhevsky, I. Sutskever, and G. Hinton. "ImageNet Classification with Deep Convolutional Neural Networks." In: *Neural Information Processing Systems* 25 (Jan. 2012). DOI: 10.1145/3065386.

[44] W. H. Kruskal and W. A. Wallis. "Use of ranks in one-criterion variance analysis." In: *Journal of the American statistical Association* 47.260 (1952), pp. 583–621.

[45] D. Kucharski, P. Kleczek, J. Jaworek-Korjakowska, G. Dyduch, and M. Gorgon. "Semi-Supervised Nests of Melanocytes Segmentation Method Using Convolutional Autoencoders." In: *Sensors* 20.6 (2020). ISSN: 1424-8220. DOI: 10.3390/s20061546. URL: https://www.mdpi.com/1424-8220/20/6/1546.

[46] Z. Lai and H. Deng. "Medical Image Classification Based on Deep Features Extracted by Deep Model and Statistic Feature Fusion with Multilayer Perceptron." In: *Computational Intelligence and Neuroscience* 2018 (Sept. 2018), pp. 1–13. DOI: 10.1155/2018/2061516.

[47] Z. Liu, Y. Cao, Y. Li, X. Xiao, Q. Qiu, M. Yang, Y. Zhao, and L. Cui. "Automatic Diagnosis of Fungal Keratitis Using Data Augmentation and Image Fusion with Deep Convolutional Neural Network." In: *Computer Methods and Programs in Biomedicine* 187 (Aug. 2019), p. 105019. DOI: 10.1016/j.cmpb.2019.105019.

[48] L. Van der Maaten and G. Hinton. "Visualizing data using t-SNE." In: *Journal of machine learning research* 9.11 (2008).

[49] J. Masci, U. Meier, D. Ciresan, and J. Schmidhuber. "Stacked Convolutional Auto-Encoders for Hierarchical Feature Extraction." In: vol. 9. June 2011, pp. 52–59. ISBN: 978-3-642-21734-0. DOI: 10.1007/978-3-642-21735-7_7.

[50] D. Masters and C. Luschi. "Revisiting small batch training for deep neural networks." In: *arXiv preprint arXiv:1804.07612* (2018).

[51] A. Maćkiewicz and W. Ratajczak. "Principal components analysis (PCA)." In: *Computers Geosciences* 19.3 (1993), pp. 303–342. ISSN: 0098-3004. DOI: https://doi.org/10.1016/0098-3004(93)90090-R. URL: https://www.sciencedirect.com/science/article/pii/009830049390090R.

[52] M. Naderan and Y. Zaychenko. "Convolutional Autoencoder Application for Breast Cancer Classification." In: Oct. 2020, pp. 1–4. DOI: 10.1109/SAIC51296.2020.9239139.

[53] Y. Newton, A. Novak, T. Swatloski, D. McColl, S. Chopra, K. Graim, A. Weinstein, R. Baertsch, S. Salama, K. Ellrott, M. Chopra, T. Goldstein, D. Haussler, O. Morozova, and J. Stuart. "TumorMap: Exploring the Molecular Similarities of Cancer Samples in an Interactive Portal." In: *Cancer Research* 77 (Nov. 2017), e111–e114. DOI: 10.1158/0008-5472.CAN-17-0580.

[54] O. Oktay, J. Schlemper, L. Folgoc, M. Lee, M. Heinrich, K. Misawa, K. Mori, S. McDonagh, N. Hammerla, B. Kainz, B. Glocker, and D. Rueckert. "Attention U-Net: Learning Where to Look for the Pancreas." In: (Apr. 2018).

[55] J. Park, K. Kim, Y. Nam, M. Choi, S. Choi, and J. Rhie. "Convolutional-neural-network-based diagnosis of appendicitis via CT scans in patients with acute abdominal pain presenting in the emergency department." In: *Scientific Reports* 10 (June 2020). DOI: 10.1038/s41598-020-66674-7.

[56] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. "Scikit-learn: Machine Learning in Python." In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[57] H. Pratt, F. Coenen, D. M. Broadbent, S. P. Harding, and Y. Zheng. "Convolutional Neural Networks for Diabetic Retinopathy." In: *Procedia Computer Science* 90 (2016). 20th Conference on Medical Image Understanding and Analysis (MIUA 2016), pp. 200–205. ISSN: 1877-0509. DOI: https://doi.org/10.1016/j.procs.2016.07.014. URL: https://www.sciencedirect.com/science/article/pii/S1877050916311929.

[58] T. Rahman, L. Mahanta, C. Chakraborty, A. DAS, and J. SARMA. "Textural pattern classification for oral squamous cell carcinoma." In: *Journal of Microscopy* 269 (Aug. 2017). DOI: 10.1111/jmi.12611.

[59] A. Rosebrock. *Deep Learning for Computer Vision with Python*. pyimagesearch, 2018. URL: https://books.google.pt/books?id=60IvygEACAAJ.

[60] M. Roser and H. Ritchie. "Cancer." In: *Our World in Data* (2015). https://ourworldindata.org/cancer.

[61] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. "Learning representations by back-propagating errors." In: *Nature* 323 (Oct. 1986), pp. 533–536. DOI: 10.1038/323533a0.

[62] M. Serag, M. Wael, I. Yassine, and A. Fahmy. "Cardiac MRI View Classification using Autoencoder." In: Dec. 2014. DOI: 10.1109/CIBEC.2014.7020935.

[63] T. Shen, T. Zhou, G. Long, J. Jiang, S. Wang, and C. Zhang. "Reinforced Self-Attention Network: a Hybrid of Hard and Soft Attention for Sequence Modeling." In: July 2018, pp. 4345–4352. DOI: 10.24963/ijcai.2018/604.

[64] J. Tan, M. Ung, C. Cheng, and C. Greene. "Unsupervised feature construction and knowledge extraction from genome-wide assays of breast cancer with denoising autoencoders." In: *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing* 20 (Jan. 2015), pp. 132–43. DOI: 10.1142/9789814644730_0014.

[65] R. Thawani, M. McLane, N. Beig, S. Ghose, P. Prasanna, V. Velcheti, and A. Madabhushi. "Radiomics and Radiogenomics in Lung Cancer: A Review for the Clinician." In: *Lung Cancer* 115 (Nov. 2017). DOI: 10.1016/j.lungcan.2017.10.015.

[66] C. Vasconcelos and B. Vasconcelos. "Increasing Deep Learning Melanoma Classification by Classical And Expert Knowledge Based ImageTransforms." In: (Feb. 2017).

[67] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, and X. Tang. "Residual Attention Network for Image Classification." In: July 2017, pp. 6450–6458. DOI: 10.1109/CVPR.2017.683.

[68] J. Wang, Y. Li, Y. Zhang, H. Xie, and C. Wang. "Bag-of-Features Based Classification of Breast Parenchymal Tissue in the Mammogram via Jointly Selecting and Weighting Visual Words." In: *2011 Sixth International Conference on Image and Graphics*. 2011, pp. 622–627. DOI: 10.1109/ICIG.2011.192.

[69] Y. Wang, L. Sun, K. Ma, and J. Fang. "Breast Cancer Microscope Image Classification Based on CNN with Image Deformation." In: June 2018, pp. 845–852. ISBN: 978-3-319-92999-6. DOI: 10.1007/978-3-319-93000-8_96.

[70] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon. "CBAM: Convolutional Block Attention Module." In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018.

[71] B. Xu, J. Liu, X. Hou, B. Liu, J. Garibaldi, I. Ellis, A. Green, L. Shen, and G. Qiu. "Attention by Selection: A Deep Selective Attention Approach to Breast Cancer Classification." In: *IEEE Transactions on Medical Imaging* 39 (2020), pp. 1930–1941.

[72] B. Xu, J. Liu, X. Hou, B. Liu, J. Garibaldi, I. Ellis, A. Green, L. Shen, and G. Qiu. "Look, Investigate, and Classify: A Deep Hybrid Attention Method for Breast Cancer Classification." In: Apr. 2019, pp. 914–918. DOI: 10.1109/ISBI.2019.8759454.

[73] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhutdinov, R. Zemel, and Y. Bengio. "Show, Attend and Tell: Neural Image Caption Generation with Visual Attention." In: (Feb. 2015).

[74] L. Xu, J. Huang, A. Nitanda, R. Asaoka, and K. Yamanishi. "A Novel Global Spatial Attention Mechanism in Convolutional Neural Network for Medical Image Classification." In: *ArXiv* abs/2007.15897 (2020).

[75] G. Xue, S. Liu, and Y. Ma. "A hybrid deep learning-based fruit classification using attention model and convolution autoencoder." In: *Complex Intell Systems* 2020 (Oct. 2020). DOI: 10.1007/s40747-020-00192-x.

[76] S. Yadav and S. Jadhav. "Deep convolutional neural network based medical image classification for disease diagnosis." In: *Journal of Big Data* 6 (Dec. 2019). DOI: 10.1186/s40537-019-0276-2.

[77] C. Yan, J. Yao, R. Li, Z. Xu, and J. Huang. "Weakly Supervised Deep Learning for Thoracic Disease Classification and Localization on Chest X-rays." In: *Proceedings of the 2018 ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics* (2018).

[78] F. Yang, H. Lu, W. Zhang, and G. Yang. "Visual tracking via bag of features." In: *Image Processing, IET* 6 (Mar. 2012), pp. 115–128. DOI: 10.1049/iet-ipr.2010.0127.

[79] C. Zhang, X. Cheng, J. Liu, J. He, and G. Liu. "Deep Sparse Autoencoder for Feature Extraction and Diagnosis of Locomotive Adhesion Status." In: *Journal of Control Science and Engineering* 2018 (July 2018), pp. 1–9. DOI: 10.1155/2018/8676387.

[80] Y. Zhou, J. Xu, Q. Liu, C. Li, Z. Liu, M. Wang, H. Zheng, and S. Wang. "A Radiomics Approach With CNN for Shear-Wave Elastography Breast Tumor Classification." In: *IEEE Transactions on Biomedical Engineering* 65.9 (2018), pp. 1935–1942. DOI: 10.1109/TBME.2018.2844188.