FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Image Modality Classification in Dermatology

Ana Catarina Falcão Morgado

DISSERTATION WORK

INTEGRATED MASTER IN BIOENGINEERING

Supervisor at FEUP: **Luís Filipe Teixeira, PhD**

Supervisor at Fraunhofer Portugal: **Maria Vasconcelos, PhD**

July 23, 2021

Image Modality Classification in Dermatology

Ana Catarina Falcão Morgado

INTEGRATED MASTER IN BIOENGINEERING

July 23, 2021

Resumo

Atualmente, o cancro da pele é um dos tipos de cancro mais prevalente em todo o mundo, nomeadamente nas populações de pele clara, consistindo num problema para os serviços de saúde. Dado o aspeto visual das lesões cutâneas, a teledermatologia tem vindo a permitir uma melhoria da qualidade da prestação de cuidados médicos nesta área a toda a população, caraterizando-se pela aquisição de imagens que são armazenadas e enviadas a um dermatologista de referência. Com base nas orientações estabelecidas para as consultas teledermatológicas, é possível distinguir estas imagens em cinco modalidades diferentes: anatómica, dermoscópica, de corpo inteiro, macroscópica, e também relatórios clínicos. Em algumas situações, as imagens adquiridas podem mesmo incluir uma régua junto à lesão, permitindo ao médico inferir o seu tamanho.

Dado o aumento crescente que todos os anos é verificado nos registos médicos, sistemas automáticos capazes de diferenciar as imagens de acordo com as suas modalidades e atributos (como a presença de uma régua) podem ser essenciais para que estes sejam melhor organizados. Visto que os dados médicos estão sempre a evoluir, estes sistemas precisam de ser continuamente treinados, permitindo a sua adaptação a novas condições sem a necessidade de recorrer a toda a informação previamente disponível. No entanto, treinar modelos de formal incremental está geralmente associado a uma questão designada por esquecimento catastrófico, que consiste na diminuição do desempenho relativamente aos conhecimentos anteriormente adquiridos. Assim, embora nos últimos anos tenham sido feitos alguns esforços de modo a permitir que os modelos sejam progressivamente treinados, poucos estudos foram realizados em contexto médico. Por este motivo, a necessidade de desenvolver algoritmos neste sentido mantém-se, especialmente no caso da dermatologia, podendo contribuir para uma otimização dos processos teledermatológicos entre as unidades de cuidados primários e os serviços de dermatologia.

Tendo isto em conta, esta dissertação compreendeu dois grandes objetivos: o desenvolvimento e implementação de vários algoritmos de classificação e de deteção de objetos, a fim de verificar qual a melhor abordagem para prever se uma régua estava contida em imagens dermatológicas; e o desenvolvimento de modelos capazes de classificar com precisão imagens dermatológicas de acordo com sua modalidade, os quais devem utilizar diferentes estratégias de aprendizagem incremental com o intuito de permitir o seu treino contínuo.

Na primeira parte do trabalho e em relação aos algoritmos de classificação, foram explorados três modelos diferentes: uma simples CNN treinada de raiz, uma rede VGG-16 pré-treinada na base de dados ImageNet e ajustada a este problema binário (*fine-tuned*), e uma rede VGG-16 pré-treinada na mesma base de dados e utilizada como extrator de caraterísticas. Foram feitos diferentes estudos, nomeadamente em relação à *loss function* utilizada, a necessidade de aumento de dados, e a influência da classe dermoscópica no desempenho dos algoritmos. Todos os modelos alcançaram melhores resultados quando uma *loss function* ponderada foi considerada, e apenas o *fine-tuned* VGG-16 beneficiou de um aumento do número de imagens durante o processo de aprendizagem. Relativamente à classe dermoscópica, verificou-se que os três modelos foram capazes de alcançar melhores resultados quando esta foi tida em consideração. No que respeita aos algo-

ritmos de deteção de objetos, foram considerados três detetores: o EfficientDet-D0, o RetinaNet, e o Faster R-CNN, que foi o detetor capaz de obter os melhores resultados. No caso da RetinaNet, foram utilizadas duas redes de backbone: uma ResNet-50 e uma ResNet-101, não tendo sido verificadas diferenças relevantes nos resultados obtidos por ambas. Em geral, os algoritmos de classificação revelaram-se mais eficientes do que os de deteção de objetos na resolução desta tarefa. O *fine-tuned* VGG-16 foi o modelo de classificação capaz de obter os melhores resultados com uma precisão de 0.993, ultrapassando o valor de precisão de 0.925 alcançado pelo detetor Faster R-CNN.

Relativamente ao outro objetivo deste trabalho, começaram por ser desenvolvidos dois modelos capazes de classificar com precisão imagens dermatológicas de acordo com a sua modalidade: uma arquitetura VGG-16 e uma arquitectura MobileNetV2 pré-treinadas na base de dados ImageNet, o que permitiu compreender a influência da complexidade do modelo no esquecimento catastrófico. Estes modelos foram utilizados como modelos de base para o processo de aprendizagem incremental, onde foram utilizadas três estratégias diferentes de aprendizagem incremental considerando parâmetros distintos aquando da sua implementação: a *Elastic Weight Consolidation* (EWC), a *Averaged Gradient Episodic Memory* (AGEM), e *Experience Replay*, sendo a primeira uma estratégia de regularização e as duas últimas estratégias de ensaio. O treino da fase incremental foi feito considerando diferentes números de *epochs* e verificou-se que, à exceção do modelo VGG-16 utilizando a estratégia AGEM com um tamanho de memória de 50 e a estratégia Experience Replay no mesmo modelo, para todas as outras estratégias, à medida que o número de *epochs* aumentava, o desempenho dos modelos diminuía, levando a um maior esquecimento dos conhecimentos anteriormente aprendidos. Além disso, comparando o desempenho destas estratégias nos dois modelos, a MobileNetV2 superou o modelo VGG-16, sendo capaz de preservar mais informação relativa à primeira tarefa. Ainda, a estratégia Experience Replay foi a que proporcionou os melhores resultados, tanto em termos de precisão global como de esquecimento. A eficiência dos modelos em termos de tempo de treino e memória RAM necessária foi também avaliada: a Experience Replay no caso do modelo VGG-16 foi a que levou mais tempo a ser treinada e a estratégia AGEM demonstrou ser a que exigiu mais memória RAM durante o processo de treino em ambos os modelos.

Abstract

Nowadays, skin cancer is one of the most prevalent types of cancer worldwide, namely in fair-skinned populations, consisting of a problem for the healthcare services. Due to the visual appearance of skin lesions, teledermatology has enabled an improved quality of the medical care provision to all population, comprising the acquisition of medical images that are stored and forwarded to a reference dermatologist. Based on the established guidelines for teledermatological consultations, it is possible to distinguish these images across five different categories: anatomic, dermoscopic, full-body, macroscopic, and also clinical reports. In some situations, the acquired images may even comprise a ruler next to the lesion, allowing the physician to infer its size.

Since medical records undergo an increased growth every year, automatic systems able to differentiate images according to their modalities and attributes (such as the presence of a ruler) may be essential for a better organization of the records. As medical data is always evolving, these systems need to be continuously trained, allowing their adaptation to new conditions without resorting to all of the already seen information. Nevertheless, training models incrementally is usually prone to catastrophic forgetting, an issue that consists of a decrease on the performance concerning the previously acquired knowledge. Hence, although some effort has been done in the last years in order to allow models to be incrementally trained, only a few studies were performed in medical context. For this reason, the requirement for algorithms in this sense, especially in the case of dermatology, remains, allowing the optimization of the teledermatological processes between the primary care units and the dermatology services.

Taking this into account, this dissertation comprised two major goals: the development and implementation of several classification and object-detection algorithms in order to verify the best approach in predicting whether a ruler was contained in dermatological images; and the development of models able to accurately classify dermatological images according to their modality, which should employ different incremental learning strategies in order to allow their continuous training.

In the first part of the work and with respect to the classification algorithms, three different models were explored: a simple CNN trained from scratch, a VGG-16 network pre-trained on the ImageNet database and fine-tuned to this binary problem, and a pre-trained VGG-16 network pre-trained on the same database and used as a feature extractor. Different studies were made, namely concerning the employed loss function, the need for data augmentation, and the influence of the dermoscopic class on the algorithms performance. All models achieved better results when a weighted cross-entropy loss function was considered, and only the fine-tuned VGG-16 benefited from an improved amount of images during the training process. With respect to the dermoscopic class, it was verified that the three models were able to achieve better results when this class was considered. Regarding the object-detection algorithms, three detectors were considered: EfficientDet-D0, RetinaNet, and Faster R-CNN, with the latter obtaining the best results. In the case of RetinaNet, two different backbone networks were employed: a ResNet-50 and a ResNet-101, and no marked differences were found in the results obtained by the two networks.

In general, the classification algorithms proved to be more efficient than the object-detection ones in solving this task, being the fine-tuned VGG-16 the classification model that provided the best outcomes with an accuracy of 0.993, surpassing the 0.925 accuracy value achieved by the Faster R-CNN.

Concerning the other goal of this work, two models that could classify dermatological images according to their modality were firstly developed: a VGG-16 and a MobileNetV2 architecture pre-trained on the ImageNet database, which allowed to understand the influence of the model's complexity in the catastrophic forgetting. These models were used as the base models for the incremental learning process, where three different incremental learning strategies considering distinct parameters upon their implementation were employed: the Elastic Weight Consolidation (EWC), the Averaged Gradient Episodic Memory (AGEM), and the Experience Replay, being the first a regularization strategy and the last two rehearsal strategies. The training of the incremental phase was made considering different numbers of epochs and it was verified that, with the exception of the VGG-16 model employing the AGEM strategy with a memory size of 50 and the Experience Replay strategy, for all other strategies, as the number of epochs increased, the performance of the models decreased, leading to a higher forgetting of the previously learned knowledge. Also, comparing the performance of these strategies on the two models, the MobileNetV2 outperformed the VGG-16 model, being able to preserve more information concerning the first task. Moreover, the Experience Replay strategy was the one that provided the best outcomes both in terms of the global accuracy and forgetting. The efficiency of the models in terms of training time and computation was also assessed: the Experience Replay in the case of the VGG-16 model was the one that took longer to be trained and the AGEM strategy demonstrated to be the one that required more RAM memory during the training process for both models.

Agradecimentos

Em primeiro lugar, gostaria de agradecer à Fraunhofer pela oportunidade de desenvolver a minha dissertação, permitindo-me ter uma perceção diferente acerca do mundo da investigação (embora que virtual).

De seguida, as minhas palavras de agradecimento são dirigidas aos meus orientadores, Dra. Maria Vasconcelos e Professor Luís Teixeira por estarem sempre disponíveis para me ouvir e por me terem guiado e apoiado neste desafio. À Catarina Andrade, por todas as sugestões que me foi dando ao longo da realização deste trabalho, e por estar sempre pronta para esclarecer as minhas dúvidas.

Como nem só de trabalho se fez este percurso, quero agradecer a todos os meus amigos que partilharam comigo estes cinco anos ou parte deles. Começando pelo "Dutch Gang", obrigada Cat, Inês, Ritinha, Miguel e Mariana por todas as aventuras que começaram no jardim da FEUP e chegaram além-fronteiras. Um agradecimento especial às minhas "Babes de Twente" pelos 6 meses incríveis que passamos no Z11 que, sem dúvida, vão ficar para sempre guardados. E como é impossível falar do Z11 sem falar do Pedro, obrigada também a ti por teres sido o melhor inquilino e nos teres preparado para o que se avizinhava neste ano: a procura por resultados. Obrigada ao grupinho da "Salvação", Rafa, Patrícia, Margarida e Inês pela companhia nas horas de almoço e nas aulas mais aleatórias. Como é óbvio, não podia deixar de agradecer à Bia, Sofia e Maria Joo por, apesar de termos seguido por caminhos diferentes, não ter deixado de haver pontos de encontro para conversas, desabafos e brincadeiras.

Mas como nada disto teria sido a mesma coisa sem o apoio incansável dos meus pais e do meu irmão que, ao longo destes 5 anos, estiveram sempre do meu lado e tiveram a paciência necessária para me aturar nas horas de maior stress, deixo-lhes aqui o meu agradecimento. Nem mil "obrigadas" seriam suficientes para lhes retribuir tudo o que fizeram por mim!

Ana Catarina Morgado

"Research means that you don't know,
but are willing to find out."

Charles F. Kettering

Contents

List of Figures

List of Tables

Abbreviations

AAD	American Academy of Dermatology Association
AGEM	Averaged Gradient Episodic Memory
AODE	Averaged One Dependence Estimator
AK	Actinic Keratosis
ASPP	Atrous Spatial Pyramid Pooling
ATA	American Telemedicine Association
AUC	Area Under Curve
BCC	Basal Cell Carcinoma
BiFPN	Bidirectional Feature Pyramid Network
BN	Batch Normalization
BNC	Bayesian Network Classifiers
BoK	Bag of Keypoints
BoW	Bag-of-Words
BOVW	Bag-of-Visual-Words
BoC	Bag-of-Colors
BWT	Backward Transfer
CEDD	Color and Edge Directivity Descriptor
CNN	Convolutional Neural Network
CT	Computed Tomography
DGS	Direção-Geral da Saúde
DNN	Deep Neural Network
EWC	Elastic Weight Consolidation
FCH	Fuzzy Color Histogram
FCTH	Fuzzy Color and Texture Histogram
FIFO	First-In-First-Out
FN	False Negatives
FP	False Positives
FPR	False Positive Rate
FWT	Forward Transfer
GEM	Gradient of Episodic Memory
GIFT	GNU Image-Finding Tool
IoU	Intersection over Union
ISIC	International Skin Imaging Collaboration
k-NN	k-Nearest Neighbors
LBP	Local Binary Patterns
LDA	Linear Discriminant Analysis
LwF	Learning without Forgetting
MB	MegaBytes
MKL	Multiple Kernel Learning
ML	Machine Learning

MN	Melanocytic Nevi
NMSCs	Non-melanoma Skin Cancers
NN	Neural Networks
NPD	Non-polarized Dermoscopes
PReLU	Parametric ReLU
PCA	Principal Component Analysis
PD	Polarized Dermoscopes
PHOW	Pyramid Histogram of Visual Words
PNN	Progressive Neural Networks
QBoC	Quad-Tree Bag-of-Colors
ReLU	Rectified Linear Unit
ROC	Receiver Operating Characteristic
RoI	Regions of Interest
SAF	Store-and-Forward
SCC	Squamous Cell Carcinoma
SDL	Synergic Deep Learning
SGD	Stochastic Gradient Descent
SI	Synaptic Intelligence
SIFT	Scale-invariant Feature Transform
SK	Seborrheic Keratosis
SL	Solar Lentigines
SPP	Spatial Pyramid Pooling
SVM	Support-vector Machine
TBP	Total-body Photography
TDS	Total Dermoscopy Score
TN	True Negatives
TP	True Positives
TPR	True Positive Rate
UV	Ultraviolet
VSM	Vector Space Model

Chapter 1

Introduction

1.1 Context and Motivation

Skin cancer is the most frequent malignancy in fair-skinned populations with a worldwide increasing incidence [1][2]. Skin cancer term is widely used to refer to any malignant lesion occurring in the skin, comprising disorders as Basal Cell Carcinoma (BCC), Squamous Cell Carcinoma (SCC), or Melanoma [1].

BCC and SCC, which are also known as non-melanoma skin cancers (NMSC), present an incidence much higher comparing to melanoma and other types of cancer, growing up to 10% every year [3]. Only in 2020, more than 1 million new cases were diagnosed worldwide, representing an increasing problem to the health services due to the impact on the healthcare costs and limited resources to respond to all patients [4][5][6]. Among NMSCs, BCC is the most common disease, corresponding to approximately 80% of all keratinocyte carcinomas, followed by SCC which represents almost 20% of these cancers [1].

Regarding the cutaneous Melanoma, an annual increase of around 3 to 8% over the past decades in Europe, the United States, Australia, and Canada was estimated [7][8]. In 2020, nearly 300 thousand new diagnoses were reported worldwide, a value much lower than the new NMSCs cases. Although Melanoma is less common, it is much more lethal, having been reported around 60 thousand deaths in 2020 due to this disease [4].

On the one hand, if the detection of skin cancers only occurs when the malignant cancer has already spread to other body organs distant to the origin site, the five-year relative survival rate can drop from 92% to 25% [9]. On the other hand, when diagnosed in the earliest stages, most skin carcinomas can be easily treated [10]. For this reason, a timely and accurate diagnosis of skin lesions is clinically important to improve the prognosis of patients [11].

As skin cancer diagnosis is typically performed by visual inspection, it can be a very challenging task due to the similarities between the most common types of skin cancers and benign lesions. Besides, this inspection is time-consuming and may take to subjective results. Hence, dermatologists usually resort to auxiliary techniques, such as dermoscopy, to assist noninvasive diagnosis of skin disorders and improve its accuracy [11][12][13]. Dermoscopy is a standard procedure for

skin imaging, namely in the case of melanocytic lesions, that magnifies both surface and subsurface structures, making them perceptible to the naked eye [12]. Although this technique enhances the precision of the diagnosis, the corresponding learning curve is steep, requiring a lot of training to be properly used [13]. Thus, in most cases, dermoscopy is only used by experienced dermatologists, being proved to have no relevant efficiency when used by less experienced dermatologists [11].

Therefore, it is essential that technologies able to aid dermatologists dealing with the diagnosis of skin lesions, avoiding the misdiagnosis of skin cancer, and anticipating its detection are developed. This is where computer-aided systems may play a relevant role, being increasingly used to assist physicians in medical decision-making [14]. In the field of dermatology, for instance, deep (and machine) learning approaches have already shown their effectiveness in skin lesions diagnosis, achieving equal or even better results comparing to dermatologists [15][16][17].

Nevertheless, the automatic classification of lesions through these models also presents its limitations. Training deep learning models involves a large amount of labeled data so they can be applied to real-world problems. However, storing this amount of data may be expensive in terms of the required memory and besides, the performance of these algorithms depends on the acquisition properties of the trained images as well. In dermatology, the images used to feed deep learning algorithms may come from teledermatological consults, in which the primary care physician takes photos of the patient's skin and forward them to a referenced dermatologist who will then analyze and evaluate them. To facilitate the appraisal, in some situations, the acquired images may even comprise a ruler next to the lesion, allowing the physician to infer its size.

Due to the improvements in medical imaging equipments, there has been an increased use of teledermatology in the last years which contributes to the annual growth of around 20% to 40% that is verified in the amount of acquired medical images [18]. This may represent a problem for the organization of medical records since their categorization is mainly done manually, which is time-consuming, and prone to errors. Taking this into account, the access to specific clinical information may be demanding, and to tackle this problem, some clinicians have even assumed that imaging modality was one of the most important filters to improve the search among medical records, as it considers visual characteristics of images [19]. However, the existing databases are not always categorized or filtered in this sense [18][20].

In terms of dermatology, according to the established guidelines [21], it is possible to categorize the images acquired for teledermatological consults in different modalities. Having in mind that different equipments are used to obtain these images, they may differ in terms of their visual appearance and properties. Thus, a system able to previously classify the received images according to their modalities and attributes (such as the presence of a ruler) could be an asset to better organize the existing databases and consequently improve the already implemented diagnosis algorithms. Moreover, as medical data is always evolving, these systems need to be updated as new images are available. Nevertheless, on the one hand, the previously trained images may not be available anymore due to memory issues and, on the other hand, training large networks from scratch demands high computational cost and energy. Therefore, algorithms able to incrementally

be trained are being required more and more, avoiding the access to all of the already seen images and allowing models to adapt to new information [22].

1.2 Research Goals

Considering what was previously mentioned, this dissertation comprises two main goals.

As the clinical images acquired by physicians for teledermatological consults may contain a ruler next to the lesion to later infer its size, the first goal of this work consists of developing and implementing several classification and object-detection algorithms in order to predict whether a ruler is contained in an image. Also, the performance of these two types of methodologies should be compared to evaluate which is the best approach to find the presence of this object in dermatological images.

This dissertation also aims to accurately classify different modalities of skin images, namely anatomic, dermoscopic, full-body, macroscopic images, and clinical reports. To accomplish this goal, several machine or deep learning approaches should be explored. Furthermore, in order to allow these models to be continuously trained and to prevent that all images are maintained in the memory after training, incremental learning techniques should be implemented allowing them to be updated as new information is available without the need of retraining the entire network with all data, nor forgetting what was previously learned. Taking this into account, it is expected to overcome some of the limitations presented in the previous section.

1.3 Expected Contributions

This work is part of the "Derm.AI: Usage of Artificial Intelligence to Power Teledermatological Screening" project carried out by Fraunhofer Portugal Research Centre (FhP-AICOS), with reference DSAIPA/AI/0031/2018, and supported by national funds through 'FCT—Foundation for Science and Technology, I.P.". The major goal of Derm.AI project is to contribute to the optimization of Teledermatology processes between Primary Care Units and Dermatology Services of the National Health Service, through the integration of a mobile application to acquire macroscopic skin lesion images and the development of AI-powered Risk Prioritization and Decision Support platform [23]. Hence, the ultimate contribution of this dissertation consists of a better organization of the medical records in the dermatological field.

To achieve this final aim, the expected contributions of this work comprise:

- The implementation and comparison of different classification and object-detection algorithms able to predict the presence of a ruler in different types of dermatological images.
- An annotated dataset concerning the localization of rulers in dermatological images.
- The development and comparison of several algorithms able to accurately classify the modalities of different dermatological images.

- The implementation and comparison of different incremental learning strategies to allow the continuous training of the developed classification algorithms.

1.4 Document Structure

This document is composed of six chapters. The current chapter (Chapter 1) introduces the context and motivation behind the work, the goals that are intended to be achieved and the contributions that are expected to be provided. For a better comprehension of the developed work, Chapter 2, *Background*, comprises different sections that intend to address the underlying concepts of this dissertation. Thus, these sections concern skin cancer, Telemedicine and Teledermatology, the basics of machine and deep learning, object-detection algorithms and the essentials of incremental learning. The most common metrics used to evaluate the performance of the computer vision models are also introduced in this chapter. In Chapter 3, *State-of-the-Art*, an overview of the work already developed in these themes is given, namely the current methodologies found in the literature regarding medical imaging modality classification, skin lesion classification, object detection and incremental learning. The methodologies applied during this work are presented in Chapter 4, whereas the achieved results and the corresponding analysis can be found in Chapter 5. Finally, in Chapter 6, *Conclusions and Future Work*, the main conclusions achieved during the development of this dissertation are discussed, as well as some remarks about future research directions.

Chapter 2

Background

For a better comprehension of the developed work, this chapter is composed of eight different sections that intend to address the main background concepts approached throughout this dissertation. The first section (Section 2.1) is regarding skin cancer, including a description of the disease, where the different types are introduced, as well as the associated risk factors and diagnosis approaches; the following section (Section 2.2) concerns Telemedicine and Teledermatology, approaching the current guidelines for its practice; the third section (Section 2.3) involves the presentation of the mainly used medical imaging techniques, focusing on the dermatological ones; this chapter also comprises four sections dedicated to the basics of machine and deep learning (Sections 2.4 and 2.5), the object-detection algorithms (Section 2.6) and also the essentials of incremental learning (Section 2.7), respectively; the last section (Section 2.8) of this chapter presents the most common metrics used to evaluate the performance of the computer vision models.

2.1 Skin Cancer

Skin Cancer is the most prevalent type of cancer in fair-skinned populations [1][7]. Only in 2020, more than 1 million new cases of NMSCs were diagnosed worldwide, and new Melanoma diagnosis comprised more than 300 thousand cases, 1071 of which were reported in Portugal [4][24]. It is estimated that, every year, the incidence of this disease increases up to 10% [3], which poses a problem for the health systems, namely in terms of the associated costs and responsiveness [2][7]. For this reason, since the treatment of skin lesions can be relatively simple if diagnosed in the earliest stages, timely and accurate diagnosis can be the key to control this global public health issue [9].

2.1.1 Skin lesions

Skin is the external covering of the body, consisting of the largest human body organ. Since it is an interface with the environment, skin is subject to several factors that can compromise its appearance, such as infections, allergic reactions, or even the genetics itself. These alterations in the skin may consist of cutaneous lesions. The majority of skin lesions are benign and do not

require special medical cares. Nevertheless, as it will be presented in this section, regular skin inspection is essential, as benign lesions can be easily misdiagnosis as other types of more serious injuries.

2.1.1.1 Malignant Skin Lesions

Malignant skin lesions are mostly related to non-melanoma skin cancers (NMSCs), also known as keratinocyte cancers, and malignant Melanoma.

Regarding NMSCs, these are by far the most commonly diagnosed cancer type worldwide [7][25]. However, the exact number of people living with this disease is unknown, since NMSCs are not always reported in cancer registries [5][26]. For this reason, the incidence of these cancers can be 18-20 times greater than the one verified for Melanoma [2][26]. On the other hand, although the high incidence, the mortality rate associated to non-melanoma skin cancers is low [10], being the overall five-year relative survival rate of around 92% [9]. Around 80% of these cancers occurs in body areas exposed to sun, namely in the lips, nose, cheek, orbit nasolabial, ears, or in the dorsum of the hands [7].

NMSCs comprise different types of cancers, being Basal Cell Carcinoma and Squamous Cell Carcinoma the most frequent ones [1][5]. These two types of non-melanoma cancers consist of around 95% of all NMSCs [5].

Basal Cell Carcinoma (BCC)

BCC represents almost 80% of all diagnosed non-melanoma skin cancers [1]. It can be defined as a locally invasive slow-growing skin tumor [27][28] and results from the uncontrolled growth of the basal cells of the skin in the external skin's layer [29].

When promptly diagnosed and if the lesion starts to be treated in the earliest phases, BCC can be straightforwardly cured [30]. Although BCC rarely leads to the formation of metastasis (the metastatic rate varies from 0.0028% to 0.55%) this cancer results in high morbidity due to tissue invasion and destruction in areas as the face and neck [1][27][31].

Early BCCs are mainly small, translucent, or pearly, with raised telangiectatic edges. Besides the usual rodent ulcer composed by an indurated edge and ulcerated centre, other subtypes of this cancer include nodular or cystic, superficial, pigmented, basisquamous, and morphoeic [32]. Therefore, as shown in Figure 2.1, BCC can be manifested in different ways, being the nodular basal-cell carcinoma the most frequent one [1][33][34].

(a) Nodular (b) Superficial (c) Basisquamous (d) Morphoeic

Figure 2.1: Types of Basal Cell Carcinoma [35].

Moreover, BCC presents a high recurrence rate that depends on the tumor subtype [36][37], being the recurrent BCC, usually, harder to treat comparing to the primary disease [27]. People who have already developed basal cell carcinomas is also more prone to get new ones in other areas of the body [38].

Squamous Cell Carcinoma (SCC)

SCC comprises almost 20% of the NMSCs [1], being the second most common form of skin cancer [29][39]. These cancers have origin in the flat cells of the upper layer of the epidermis and are mainly associated with advancing age [7][38].

Clinically, SCC commonly appears in the form of smooth or hyperkeratotic papules or nodules, that may crust, itch or bleed to the slightest touch [29][40]. At later stages, these lesions can present central ulceration or even invade other tissues [40]. Some examples of SCC lesions may be found in Figure 2.2.

SCC can also arise from other precancerous lesions, such as Actinic Keratosis (AK), which in itself is typically a benign lesion (Section 2.1.1.2), or Bowen's disease [1][39][41], mainly caused by the excessive exposure to the sun [38]. Regarding AK, it is characterized by scaly lesions that may vary from 2 to 6mm in diameter and can be more easily recognized by touch than by visual inspection [41][40]. Bowen's disease, on the other hand, consists of a slow-growing tumor that arises in elderly skin damaged by the sun [42], and frequently appears as a well-demarcated erythematous scaling patch [42][39], as presented in Figure 2.3.

Comparing to BCC, SCC presents a higher metastatic rate, that depends on the histopathological subtype, dimension, site of the tumor, and others [41][43][44]. Concerning SCC of the lip and ear, which is a severe lesion, it is estimated a rate ranging from 10 to 25% [41], involving more complex treatments.

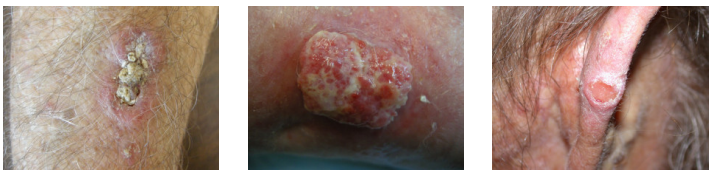
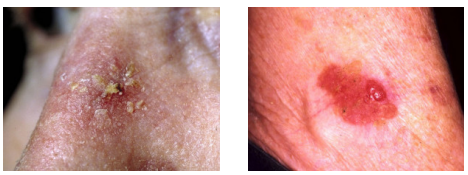


Figure 2.2: Examples of Squamous Cell Carcinoma [35].



(a) Actinic keratosis (b) Bowen's disease

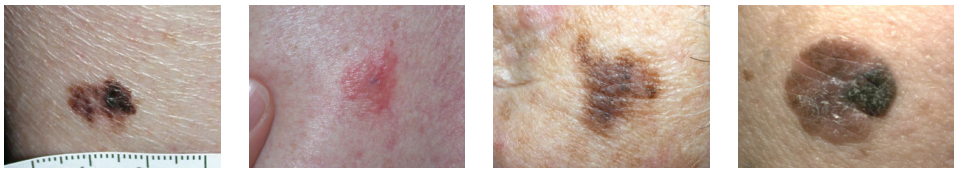
Figure 2.3: Types of SCC precancerous lesions [35].

Melanoma

Malignant Melanoma presents high morbidity and mortality, representing around 65% of all deaths related to skin cancer [45]. For this reason, although it only represents around 1% of all skin cancers, it is the most deadly one [13].

A timely diagnosis of Melanoma is, therefore, crucial for a positive prognosis of this disease, as the long-term survival rate of patients with metastasis is only 5% [45].

Melanoma arises from the uncontrolled growth of epidermal melanocytes namely due to mutations in their DNA, resulting from ultraviolet (UV) radiation exposure [29]. Although this cancer has different clinical appearances that depend on its subtype [29][46], the lesions tend to present diameter higher than 6mm, asymmetry, unequal pigmentation, and irregularities in the borders [45], as verified in Figure 2.4. Factors as the Melanoma thickness, body region, lesion's histological type, or even the patient's gender have a relevant influence on the course of the disease [47].



(a) Typical (b) Amelanotic (c) Lentigo (d) Spindle

(e) Desmosplastic (f) Nodular (g) Acral

Figure 2.4: Types of Malignant Melanoma [35].

2.1.1.2 Benign Skin Lesions

The majority of the lesions existing in skin are benign, but can be easily misdiagnosed for malignant lesions due to the similarity that often exists between them, as seen in Figure 2.5. Benign skin lesions comprise diseases as Actinic Keratosis (already introduced in Section 2.1.1.1 as a precancerous lesion), Seborrheic Keratosis, Melanocytic nevi, Solar Lentigo, Dermatofibromas, and others [48].

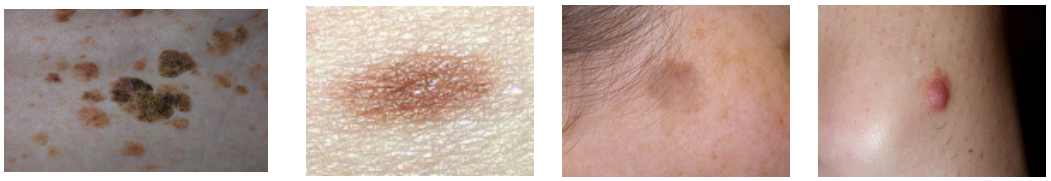
Concerning Seborrheic Keratosis (SK), this arises as sharply demarcated brown or light lesions with variable size, usually slightly elevated in the skin. SK is one of the most common benign tumors and, in some cases, may be identified as malignant Melanoma because of its similar appearance [49][50].

Melanocytic nevi (MN), also known as moles, consists of a proliferation of melanocytes and comprise different subtypes, such as congenital, acquired, blue nevi, or Spitz nevus [51]. Depending on the lesion subtype, MN presents different clinical appearances. As it will be seen in Section

2.1.2, although the risk of a specific mole become a malignant Melanoma is low [52], the amount of MN lesions may be a risk factor for the development of malignant Melanoma.

Regarding Solar Lentigo (SL), as the name suggests, this is a non-cancerous lesion that tends to appear in sun-damaged areas [48]. Clinically, SL lesions present well-defined borders, can be either light yellow or dark brown, and are variable in size [53]. These lesions are accumulated with age and affect more than 90% of Caucasian people after 50 years old [54].

Dermatofibromas may arise due to several factors and mostly affect young or middle-aged adults. These benign lesions are characterized by firm and round papules or nodules with soft surfaces that frequently appear in lower extremities. Dermatofibromas may vary from a few millimeters to centimeters and may exhibit different colors [48][55].



(a) Seborrhoeic Keratosis (b) Melanocytic nevi (c) Solar Lentigo (d) Dermatofibromas

Figure 2.5: Types of Benign Sin Lesions [35].

2.1.2 Risk factors

Different aetiology factors are associated with a higher propensity to develop skin lesions, including endogenous or exogenous conditions [44][56]. Concerning endogenous factors, phenotypic characteristics as the skin and eye color, the amount of melanocytic nevi, the existence of dysplastic nevi, and history of Melanoma in family or in the individual may influence the arise of skin lesions [56][57]. On the other hand, exogenous factors comprise exposure to UV radiation, recurrence of sunburn or care with sun protection [56]. Other factors as immunosuppression or arsenic exposure may also trigger the arising of skin lesions [27]. Although it is known that these are the main factors, their contribution to the emergence of cancer is still unclear [57]. Nevertheless, it is estimated that exposure to UV radiation is the main factor for the development of skin cancers, being associated to around 90% of all NMSCs and 67% of malignant Melanoma [30][31][58].

2.1.3 Diagnosis

As skin lesions are typically identified by their visual aspect, the first screening approach for their diagnosis consists of a visual examination of the patient's skin. Physicians should take into account the clinical history of the patient, as this may contribute to the development of different skin conditions, and evaluate the region around an identified lesion, as a secondary disease may coexist [39].

Due to the visual similarity between early-stage Melanomas, NMSCs, and benign skin lesions, the differentiation of these conditions may be a demanding task even for experienced dermatologists [59]. Being so, as it will be described in Section 2.3.1, physicians may resort to dermoscopy, an imaging technique that enables the access to deeper structures of the skin, improving the accuracy of the diagnosis [11]. However, in doubtful situations, a biopsy should still be indicated in order to perform a histologic examination [32][41], as well as additional exams, such as blood analysis, computed tomography, or magnetic resonance imaging when bony involvement or invasion of nerves are suspected [27].

Besides the common pattern analysis, which consists of a complex inspection of global and local patterns in the lesion and requires experience to be performed, some criteria were developed to aid dermatologists in the diagnosis and differentiation of similar skin disorders. Among others, these criteria comprise the ABCD rule, the 7-point Checklist, and the Menzies method which are described below [60].

ABCD(E) rule is widely used by dermatologists to assess the diagnosis of different skin lesions and to easily interpret clinical or dermoscopy images. Concerning the dermoscopy version, it relies on four criteria (A-Asymmetry, B-Border, C-Color, and D-Differential structure) that must be taken into account for the diagnosis [61]. Based on the image appearance, a score is assigned to each criterion that has an associated weight factor, as presented in Table 2.1. Therefore, it is possible to calculate the total dermoscopy score (TDS) which will correspond to the probability of the lesion being malignant. On the other hand, for clinical images, the D criterion is replaced by a *diameter greater than 6mm* and another criterion (E) is added, enabling the *evolving* of the lesion to be also considered, which results in the ABCDE rule [62].

Table 2.1: ABCD rule of dermoscopy [63].

Criteria	**Description**	**Score**	**Weight factor(q)**
Assymetry (A)	Assess the symmetry of not only the shape but also of the colors, and the structures in 0, 1, or 2 axes.	0-2	1.3
Border (B)	Evaluates the abrupt cut-off of pigment pattern border in 0-8 segments.	0-8	0.1
Colour (C)	Analyses the number of existing colors (white, red, light-brown, dark-brown, blue-gray, black). Each colour corresponds to 1 point.	1-6	0.5
Differential structure (D)	Analyses the presence of five different structures (structureless or homogeneous areas, streaks, dots, and globules). Each structure corresponds to 1 point.	1-5	0.5

$TDS = (A_{score} * A_q) + (B_{score} * B_q + (C_{score} * C_q) + (Dscore * D_q)$
$TDS < 4.55$, benign melanocytic lesion; $4.8 < TDS < 5.45$ - suspicious lesion; $TDS > 5.45$ - highly suspicious for Melanoma lesion

Concerning the **7-point checklist** approach, this intends to characterize three major, and four minor criteria on skin lesions, which are detailed in Table 2.2. These criteria also have different associated scores that allow to assess skin lesions diagnosis. If the obtained score exceeds a given threshold, the lesion is diagnosed as Melanoma [64][65]. The **Menzies method** (Table 2.3), on the other hand, comprises a total of eleven features (nine positives and two negatives) and the score is assigned depending on they are present or absent [66][67].

Table 2.2: 7-point Checklist criteria [65].

Major Criteria (Score=2)	**Minor criteria** (Score=1)
Atypical pigmented network	Irregular streaks
Blue-whitish veil	Irregular pigmentation
Atypical vascular pattern	Irregular dots/globules
	Regression structures

* Total score<3 benign lesion; Total score>=3 malignant lesion

Table 2.3: Menzies method criteria [67].

Positive Features	**Negative Features**
Blue-whitish veil	
Multiple brown dots	
Pseudopods	
Radial streaming	Symmetrical pattern
Scarlike depigmentation	One Colour
Peripheral black dots/globules	
Multiple colours (5 or 6)	
Multiple blue/grey dots	
Broad pigment network	

2.2 Telemedicine and Teledermatology

Telemedicine corresponds to the usage of telecommunication technologies for medical information delivery [68]. In the past years, there has been a growing increase in the use of this clinical procedure thanks to the benefits that have been identified regarding its use [69][70]. These advantages are many, as teleconsultations proved to enhance the clinical outcomes of the patients, due to the improvements in healthcare access and delivery, and also their cost effectiveness [71].

Given the visual nature of skin problems, telemedicine is an extremely helpful resource in the field of dermatology for accurate diagnosis or suitable treatment, as clinical information can be gathered, saved, and transmitted [68][72][73]. During lesions' evaluation, primary care physicians may capture images of the patient's skin so they can ask a more specialized dermatologist for a second opinion if they are not totally convinced, or even to monitor the progression of lesions [41]. This strategy is specially important in rural areas where medical specialties may be scarce or unavailable [68]. Also, it is considered that teledermatology allows to reduce the time spent by

dermatologists in clinical decision-making, increasing the accessibility and equity of consultations in this specialty [74]. Teledermatology comprises two different approaches that take into account both visual and prior information of the patient, namely store-and-forward (SAF) and real-time consultation [71].

Regarding the firstly mentioned care delivery platform, SAF, this consists of an asynchronous consultation whereby the medical images acquired by the patient or by the primary care physician are "stored" and "forwarded" to the referring dermatologist [73]. The dermatologist is then able to evaluate these images which are typically sent in a consultation package with other patient's clinical information [71]. Although this approach has been revealed as an effective way to indicate a treatment plan and to accurately diagnose skin lesion, since there is no interaction between patients and dermatologists, as they are distant both in time and space [68], it may be difficult for the dermatologist to access more clinical information about the patient.

On the other hand, in real-time consultations patients, dermatologists, and often the primary care clinician can interact via video conference systems. Through this synchronous platform, dermatologists can obtain more detailed information about the patient, namely regarding their clinic history, which resembles more like a normal consultation [71]. Being so, patients and clinicians are separated in space but not in time, allowing a direct interaction [68].

Moreover, it is also possible to resort to hybrid teledermatology systems where both SAF and real-time consultations are provided. Although this procedure allows to overcome some of the limitations of the previously presented methodologies, this system is less common [73].

2.2.1 Practice Teledermatology Guidelines

Worldwide, different entities are responsible for establishing guidelines for the practice of telemedicine and, more specifically, teledermatology.

In Portugal, the guidelines for conducting dermatological teleconsultations are established by *Direção-Geral da Saúde* (DGS) [21]. Besides providing information about the procedure that must be followed to schedule a teleconsultation, these recommendations also include technical specifications for image acquisition that allow to maximize its quality and, therefore, provide adequate health care to the patient.

In order to be referred to a teledermatological consult, the primary care physician must evaluate the patient and find out the need for the intervention of a specialist. Therefore, if that is verified, initial medical images of the lesion must be acquired and sent by the physician to a referring hospital. In the hospital, images go through a screening process, to assess their priority. The teleconsult is then scheduled in deferred and/or real-time and is performed by a dermatologist that should inform the primary care physician about the outcomes of the evaluation [21].

In Table 2.4, a brief description of the images that should be acquired by primary clinicians depending on the type of lesion is presented. Regarding photographs of hairy areas, these should be collected at a short distance and without hairy interference, unless these are the object of consultation.

Table 2.4: Guidelines for skin lesions images acquisition in Portugal [21].

Type of lesion	Requirements
Extensive lesions	1 anatomical region photo 1 typical lesion photo 1 full body photo (eventually)
Small lesions	1 macro photo of the lesion with a ruler 1 dermatoscopy photo (eventually)
Pigmented lesions	1 macro photo of the lesion with a ruler 1 dermatoscopy photo

Besides DGS, also the American Telemedicine Association (ATA), and the American Academy of Dermatology Association (AAD) published recommendations concerning the pratice of teledermatology in America, and the University of Queensland's Centre for Online Health together with the Australasian College of Dermatologists' E-Health Committee also developed instructions for teledermatological consults in Australia [72][73] [75][76]. These three guidelines have a lot in common. As the Portuguese ones, they address clinical, technical, and administrative aspects of these consults, ensuring the quality and safety of the service, and also the privacy of the patient, namely related to the images or video transmitted in teleconsults. Besides, they also comprise barriers and challenges that a dermatologist may face in several clinical contexts, and special considerations that should be followed by the specialist during the examination of specific anatomical regions of interest, as hair-bearing skin, pigmented lesions, or mucosal lesions.

2.3 Medical imaging modalities

The advances in medical imaging technologies and equipment have led to an increased amount of collected medical images [77][78]. It is estimated that medical imaging data grows from 20% to 40% every year [18], comprising different image acquisition modalities, such as Computed Tomography (CT), X-ray, Magnetic Resonance Imaging (MRI), Ultrasound, or Microscopy [78][79], as presented in Figure 2.6.

These images consist of a source of knowledge in the fields of education, research or even to assist clinicians in medical decision making. However, due to the huge amount of data that is acquired every year, handle these repositories in order to only extract the necessary information can be a demanding task [81]. Typically, images are indexed and categorized manually using keywords, being an expensive, subjective, time-consuming, and prone to errors process [78][82]. Moreover, the retrieval of these images is mostly text-based, using the caption or associated metadata which can be inefficient since describing images with words is not always easy and may lead to a lack in their categorization [81].

Thus, identifying visual characteristics of medical images, such as their modalities can be a fundamental task to aid in the image retrieval process and in the organization of the medical

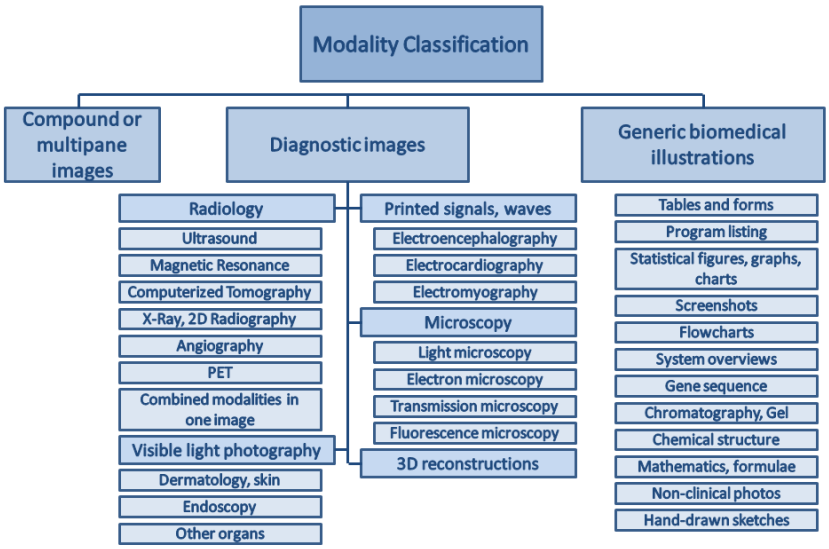


Figure 2.6: Medical Image modalities [80].

repositories [83]. To tackle this issue, modality was even defined by clinicians as one of the most relevant filters to access the desired images, as it allows to narrow the search results, improving the retrieval performance [82].

Taking this into account, the development of computer-aided systems may be an asset to make the automatic classification of image modalities [81], which will not only aid clinicians to perform more accurate diagnosis and treatments but will also be a valuable technique for healthcare students and patients [83].

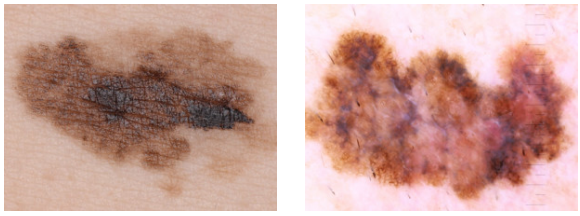
2.3.1 Dermatological imaging techniques

Due to the visual appearance of skin lesions, imaging in dermatology plays a crucial role in their diagnosis and monitoring. Similarly to other clinical specialties, also in dermatology the advances in imaging technology and the improvement of the images' quality have given rise to new acquisition techniques and also to an increase in their utilization [84]. These emerging techniques that aid dermatologists comprise digital photography, dermoscopy, optical coherence tomography, confocal microscopy, high-frequency ultrasound, and others, being the first two approaches the mostly adopted ones by physicians [85][86].

Concerning **digital photography**, this technique has been getting more and more adherence also due to the increase of teledermatological consults [84][87][88]. On the other hand, smartphones and portable cameras have been facilitating physicians to effortlessly depict the patient's skin condition. Being so, using anatomic, macroscopic, or full-body photos (modalities identified in the established teledermatology guidelines), dermatologists can track lesions over time in

follow-up consults, which allows them to make comparisons concerning the progression of cutaneous disorders [84][86]. Moreover, digital images may include total-body photography (TBP), a procedure that capture images of almost the entire patient's skin. In the traditional method, several images of the patient in different positions are taken, whereas in more recent approaches a software is used to ensemble images taken by different cameras which acquire images of the patient from different angles, being very useful for monitoring lesions' growth [73]. Therefore, digital photos have already proved to be effective in earlier diagnosis of Melanomas and other conditions in high-risk patients [89][90]. However, since common digital cameras are often used to capture images, this technique also present some limitations, namely due to the variations verified in the acquisition conditions, as the distance, or illumination. Besides, the presence of hairs, reflections or other artifacts may also influence the analysis of the lesion, as well as the image resolution [91].

Dermoscopy, as previously mentioned, is a non-invasive standard approach that enables dermatologists to access structures down to the depth of dermis that cannot be visualized at naked eye. Hence, it may contribute to the diagnosis of both pigmented and non-pigmented lesions. This technique requires the use of a dermoscope, a handheld device that allows a surface magnification up to 10x [85][92]. Dermoscopes can be divided into two categories: non-polarized dermoscopes (NPD), the more traditional ones, and polarized dermoscopes (PD) [93]. Concerning NPD, these devices require a contact medium between the skin and the device, resulting in a reduced amount of reflected light at the surface which allows to observe structures beyond it. On the other hand, PD enable the visualization of deeper skin structures avoiding the use of a liquid medium or contact with the skin [93]. Besides, these devices can be attached to a capturing device, as the smartphone, in order to acquire dermoscopic images of the cutaneous lesions of the patient. Being so, similarly to digital photos, this technique also allows to monitor skin over time as it enables to detect subtle changes in lesions [84]. Moreover, these dermoscopic images can be used in teledermatology consults, facilitating early screening of suspicious lesions [73][94]. Several studies have already shown that dermoscopy is a valuable technique for skin lesions diagnosis, improving the associated accuracy and physicians' confidence [11]. Nevertheless, as this methodology requires extensive training to be properly used, it is highly dependent on dermatologists' experience, being proved that its employment by untrained or less experienced physicians does not bring any benefits to the diagnosis [11][84].



(a) Macroscopic image (b) Dermoscopic image

Figure 2.7: Macroscopic and dermoscopic images of a superficial spreading Melanoma [35].

2.4 Machine learning

Machine learning (ML) is part of computer science, being a subfield of artificial intelligence. The main goal of ML is to build mathematical algorithms able to solve complex problems that would be hard to establish using conventional approaches. Basically, ML algorithms learn from sample data, also known as "training data", in order to create models that allow to make predictions concerning a new input [95]. Besides having the ability to learn automatically, machines can also improve their performance with the experience, without requiring explicit programming [96][97]. Hence, ML algorithms tend to be more effective than humans in solving complex problems as they are capable of finding patterns and the underlying structure in data without any bias resulting from previously acquired knowledge. Therefore, ML plays a relevant role in a wide range of areas, such as medical image analysis, or natural language processing.

Depending on the nature of the training set, ML approaches can be categorized in different paradigms, comprising supervised learning, unsupervised learning, semi-supervised learning, or reinforcement learning, being the first two the most common approaches [98].

2.4.1 Supervised Learning

In supervised learning, besides the set of data points (i.e. dataset), the algorithm is also given a ground truth associated to each example. Therefore, data is organized in pairs (x_i, y_i) that comprise both the inputs (x), which consist of feature vectors, and the desired labels (y), whereby the algorithm can learn the key characteristics that correspond to each label in order to make the predictions [96][99].

2.4.1.1 Classification vs. Regression

In a general way, supervised algorithms intend to solve one of these two main problems: classification or regression [100][101].

Classification problems aim to make a prediction of the class (label) to which the new input data belongs. Therefore, the output of a classification problem should comprise discrete values [102]. On the other hand, the purpose of the regression problems is to predict values of a continuous variable [96].

Naïve Bayes Classifier

Naïve Bayes classifier is a very popular and simple algorithm used in classification problems. This classifier relies on a probabilistic approach, namely the Bayes rule, strongly assuming the independence among features, although, in practice, this is rarely verified. Nevertheless, Naïve Bayes has been widely used for many applications, achieving promising results comparing to the state-of-the-art classifiers. The Bayes rule is represented in Equation 2.1, where $P(w|x)$ represents the probability of w, when x has already occurred; $P(x|w)$ means probability of x, when w has been verified; and $P(x)$ and $P(w)$, represent the probabilities of x and w, respectively [96].

$$P(w|x) = \frac{P(x|w)P(w)}{P(x)} \tag{2.1}$$

Logistic Regression

Logistic Regression is an example of classifier typically used for binary classification problems. The classifier employs a sigmoid function, represented in Equation 2.2, so that it is imposed that the outputs vary within the range 0 to 1. For this reason, it is possible to assume that the outputs consist of probability values, which makes logistic regression a widely used classifier. In machine learning, the variable z presented in the equation refers to a linear sum comprising unknown parameters (weights and bias) that are estimated during the learning phase [96][103].

$$g(z) = \frac{1}{1+e^{-z}} \tag{2.2}$$

Decision Trees

The idea behind decision trees relies on the "divide and conquer" approach. The classifier splits data according to a criterion in order to ensure maximum class separability, creating a tree-like structure. Typically, the information gain is taken into account as a division criterion, ensuring that at each split the decrease in entropy is maximized. The structure is composed of nodes and leaves, representing deterministic decisions. One main advantage of decision trees is the possibility of explaining the decisions based on rules, so that they are not *black-box* models [96] [104].

Support Vector Machines

Support Vector Machines (SVM) are binary classifiers whose aim is to find the hyperplane in an N-dimensional feature space that best separates data in two classes. Being so, SVMs intend to maximize the distance between data points which is given by the *margin* (Figure 2.8). Support vectors consist of the points that are closer to the hyperplane and, for that reason, influence its position and orientation. Therefore, the complexity of the classifier is not affected by the number of features in the training set but by these points [105].

For N=2, the hyperplane is a line and may be obtained by:

$$y = w.f(x) + b \tag{2.3}$$

where $f(x)$ represents the feature vector, w is the weight assigned to the corresponding feature vector, and b is a bias parameter.

In cases where data is not linearly separable and therefore a hyperplane does not exist, it is possible to map input feature vectors in a higher dimensional space, where the training set is separable. Therefore, as linear models do not work in this type of data, kernel functions that define the feature space where the classification will be performed are required [96][105].

Although SVMs are essentially used for binary classification, they can also be applied in problems involving more than two classes. Being so, by combining a number of binary classifiers it is

possible to obtain **Multiclass SVMs**. There are different strategies to handle with these problems, namely One vs. All and One vs. One approaches [106]. As the name suggests, in an One vs. All approach, each class is trained against all the others in combination, whereas in an One vs. One approach all classes are trained against each other.

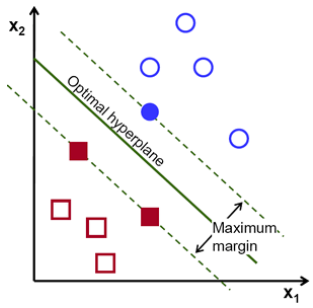


Figure 2.8: Representation of an SVM hyperplane [107].

k-Nearest Neighbours

The k-Nearest Neighbours (k-NN) consists of a supervised learning algorithm that can be both applied in classification or regression problems. This algorithm differs from the others as, instead of building a model, data is directly considered for classification without learning [104]. Thus, assuming that similar information is together, the idea of the algorithm is to find the k-nearest neighbours of a given point. The distance can be calculated by means of a distance function, such as Euclidean distance, in order to find the closest points. Therefore, in case of a classification problem, the new point is assigned with the label corresponding to the majority of its k-nearest neighbours. In a regression problem, on the other hand, the average of label values may be considered [96].

2.4.2 Unsupervised Learning

In unsupervised learning, on the other hand, the labels of the input data are not provided to the algorithm. Thereby, the learner must find out the most proper solution by itself, identifying relationships and patterns among data [96][95]. Unsupervised learning mainly comprises clustering and dimension reduction techniques [99].

2.4.2.1 Clustering

Clustering is a typical method of unsupervised learning, where it is intended to group data accordingly to patterns and similarities among data samples [97]. Hence, a clustering algorithm divides data in a fixed number of subsets, also known as *clusters*, in order to join similar input instances [101].

K-means

Being a clustering algorithm, the k-means intends to divide input data in a predefined number of clusters defined by k. The k-means assumption is that a cluster is able to represent a class. Therefore, after establishing the number of clusters there is a random initialization of k points, that correspond to the clusters' centers. For each instance of the training data, the distance to the predefined centers is calculated in order to determine which is the closest one. This represents the cluster to which the point should be assigned. When all input data is assigned to a cluster, its geometric center must be defined, corresponding to the new center of the cluster. Then, the procedure must be repeated in order to verify if any of the points should be assigned to a different cluster. The algorithm stops when it converges (i.e. when clusters' centers do not update anymore) or a predefined number of iterations is reached [96].

2.4.2.2 Dimensionality reduction

Dimensionality reduction comprises techniques that intend to reduce the number of input variables in training data, allowing to represent data with less features or lower dimensions. These techniques are very useful, as they enable a better understanding and visualization of data [99].

Principal Component Analysis (PCA)

PCA is one of the most common methods for dimension reduction, due to its simplicity and mathematical foundation. It employs simple matrix operations to map a set of correlated variables into a smaller set of linearly uncorrelated variables, preserving as much variance as possible from the original dataset [96].

2.5 Deep learning

Deep learning is a subfield of machine learning, being a valuable resource for supervised learning in complex scenarios [108]. These models allow computers to learn data representations in multiple levels of abstraction, as they are composed by several layers and units that enable to represent functions of increasing complexity [108][109]. Through backpropagation (Section 2.5.3), deep learning models teach the machine how the parameters that compute the representation of each layer should be adjusted based on the results of the previous layer. Due to its already proved effectiveness, deep learning has been changing the state-of-the-art concerning different areas, namely image and video processing, or speech recognition [109].

2.5.1 Artificial Neural Networks

Neural Networks (NN) are inspired by the human visual cortex and intend to mimic the human brain architecture, comprising several neurons connected to each other. Thus, neurons are the fundamental unit of NNs, which may receive one or many inputs (x), as represented in Figure 2.9. Through an activation function (a) of a linear transformation of the inputs, it is possible to produce

an output (y). Therefore, the output generation implies the setting of the weights associated with each input, as well as a bias parameter, followed by a non-linear operation of the corresponding weighted sum, represented by the activation function. Equation 2.4 describes this transformation, where w_k concerns the weight of the x_k input and b represents the bias [110].

$$y = a(\sum_{k=1}^{N}(w_k x_k) + b) \tag{2.4}$$

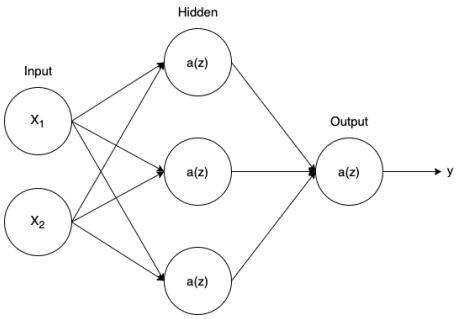
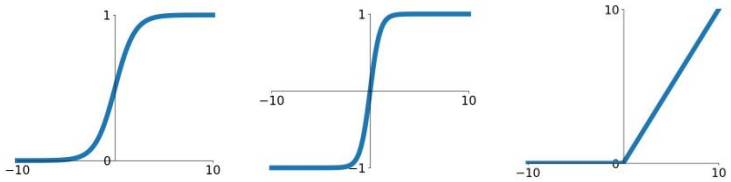


Figure 2.9: Schematic representation of a Neural Network.

Activation functions allow to introduce non-linearity along the model. The most popular ones comprise sigmoid, hyperbolic tangent (tanh), and rectified linear unit (ReLU) which are represented in Figure 2.10. Both sigmoid and tanh suffer from the vanishing gradient problem, which leads to an insignificant update in the weights that slows down the learning process. For this reason, these functions are rarely used in hidden layers. On the other hand, since the work developed by Krizhevsky *et al.* [111], the ReLU function has been gaining popularity among the activation functions. This function allows a faster learning rate due to the fact that its derivative deviates significantly from zero in positive values. Therefore, it is a valuable resource for neural networks with a high number of layers.



(a) Sigmoid (b) Tanh (c) ReLU

Figure 2.10: Examples of Activation Functions.

2.5.2 Convolutional Neural Networks

Convolution Neural Networks (CNN) have been gaining popularity within the computer vision community, proving to be a promising solution in knowledge extraction, namely for image related tasks, as object detection or image recognition [112][113].

CNNs consist of a type of feedforward neural networks since the information flows from an input x that is evaluated through an intermediate function in order to generate an output y [108]. They are composed of multiple interconnected layers containing neuron structures that are able to optimize themselves through learning [112][114]. CNN architectures are structured as a sequence of stages, that comprise different types of layers, namely convolutional and pooling for feature extraction, and fully-connected layers that perform the classification [109][115], as represented in Figure 2.11.

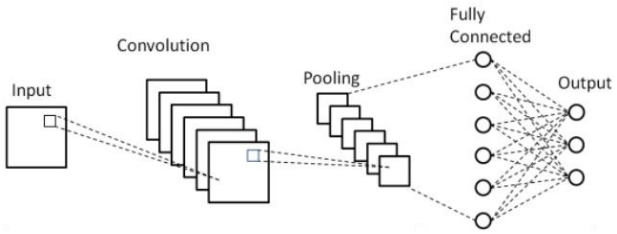


Figure 2.11: Schematic representation of a CNN [116].

2.5.2.1 Layers

Convolutional Layers

Convolutional layers (ConvLayers) are the basis of CNNs. In these layers, neurons are not connected to all neurons of the previous layer, but to a set of nodes from the preceding layer. As the name implies, ConvLayers are based on the convolution operation, presented in Figure 2.12. Hence, these layers include banks of filters with different weights organized in kernels (or filters). For each pixel on the input matrix, an element-wise product is performed with these kernels, resulting in a feature or activation map. This operation allows that input information is analyzed in a certain neighborhood, preserving important relations concerning spatial structures, that depend on the filter properties.

As the depth of the input image should be maintained throughout the layers, the number of filter channels must be in accordance, in order to generate the proper number of feature maps. Considering an RGB image of [n x n x n_d] as input, the convolution filter must have dimensions [f x f x f_d], where $f_d = n_d$. Besides filter dimensions, other parameters as stride or padding should be considered for filter design. The stride (s) concerns the size of the shift that must be performed by the sliding-window technique in the input, this is, the step between pixels at each convolution. The higher the stride, the smaller the output dimensions. Therefore, the stride is useful for dimensionality reduction. On the other hand, the padding (p) is related to the number of pixels that are added to the border of the feature map. Typically a zero padding is performed so that the convolution output preserves the initial dimensions. Wider kernels result in a higher amount of parameters and, consequently, to an increased computational cost. Taking this into account, the size of the output image can be obtained by: $[\frac{n+2p-f}{s}+1]x[\frac{n+2p-f}{s}+1]$.

Before the layer provides an output, the feature map that results from the convolution filters is processed by an activation function similar to the ones previously described.

Pooling Layers

Pooling layers are typically applied after ConvLayers and are used for down-sampling their output. Therefore, since they reduce the representation's dimensionality, they allow to decrease the computational cost. It is important to refer that, although the width of the feature maps is reduced, their depth is maintained. Depending on the kernel used for pooling, it is possible to ensemble neighbor pixels into a single value through different operations. These operations comprise average, maximum (max-pooling), minimum, or median, being the first two the most relevant ones. Being so, the kernel scans the input matrix and computes the corresponding operation on the set of selected pixels. An example of max-pooling process can be observed in Figure 2.12. In this case, pooling pixels do not overlap as it happened with convolution kernels, since typically the size of the kernel is equal to the stride [117].

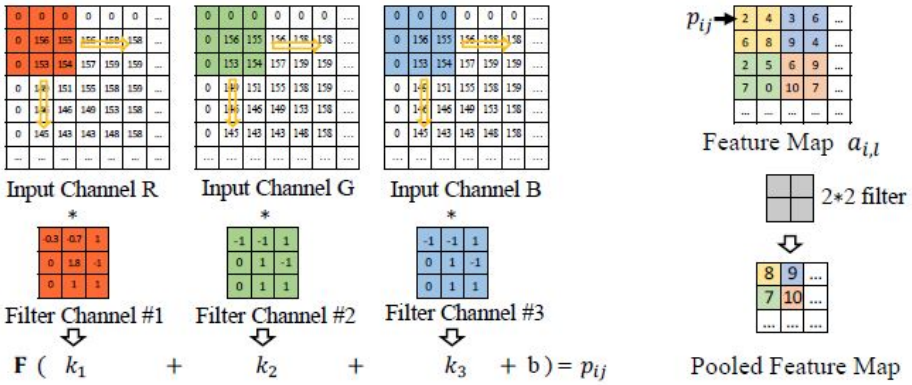


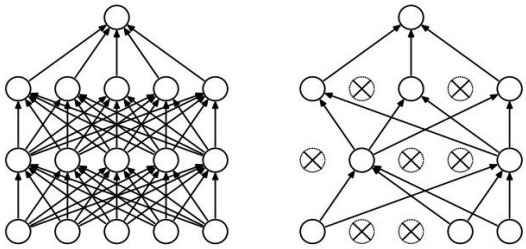
Figure 2.12: Convolutional (left) and Max-pooling processes (right) [112].

Fully-connected Layers

As the name implies, fully-connected or dense layers are directly connected to all neurons of the two adjacent layers. The inputs of these layers must be unidimensional and, for that reason, if the previous layer is a convolutional or pooling one, there is the need for flattening it into a one-dimensional vector [118]. The features extracted and down-sampled from these layers are mapped by a subset of fully-connected layers. Every fully-connected layer is followed by a nonlinear activation function, able to provide the outputs. The final layer must be adapted for the corresponding classification problem. For this reason, this layer typically contains the same number of neurons and classes, since it generates the probabilities for each class that is being predicted.

Dropout Layers

Another possible type of CNN layers are Dropout layers. These consist of a regularization technique for fully-connected layers, very useful to avoid overfitting. In these layers, a percentage of the previous layer outputs is randomly deactivated, which proved to be effective in enhancing the model's performance [119]. Although Figure 2.13 shows an example of a dropout network, the approach is similar in the case of dropout layers, as random neurons of the previous layers are set to zero.



(a) Neural Network comprising two hidden layers.

(b) Dropout network where crossed units have been dropped.

Figure 2.13: Example of Dropout Network approach [119].

2.5.2.2 Key Architectures

In the last years, several architectures and techniques have been proposed in order to improve CNNs performance. These works comprise not only deep networks such as *AlexNet* [111], *VGG* [120], *GoogleNet* [121], or *ResNet* [122], but also other approaches that allow to optimize the training of neural networks, as batch normalization [123], momentum [124], or adagrad [125]. Therefore, some of these methods will be addressed in this section.

AlexNet

In 2012, Krizhevsky *et al.* [111] proposed the AlexNet architecture within the scope of the ImageNet Scale Visual Challenge [126]. The approach comprises eight learned layers, five of which convolutional and the other three fully-connected layers, as well as three max-pooling layers, as represented in Figure 2.14. This was a revolutionary breakthrough since it consisted of one of the largest CNNs trained to the date, achieving promising results compared to the state-of-the-art methodologies. Besides the high architecture depth, the authors also considered the ReLU as the nonlinear activation functions, instead of the tanh function, and implemented a dropout technique in order to prevent overfitting.

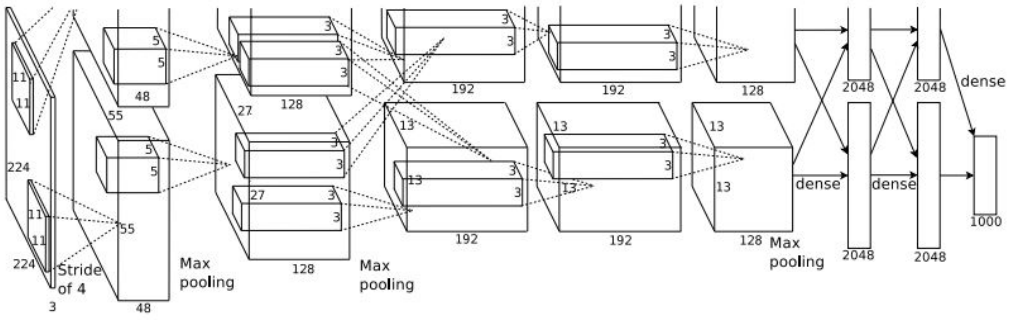


Figure 2.14: AlexNet architecture [111].

VGG

The VGG architecture was proposed by Simonyan *et al.* [120] two years after the AlexNet. The work intended to evaluate the application of small convolution filters (3x3) in increasing depth networks from 16 to 19 layers. Thus, the VGG network is composed by several convolutional layers using 3x3 kernels followed by ReLU activation functions, max-pooling layers with stride and kernel size of 2, and three fully-connected layers. Therefore, due to the reduced size of the filters used in convolutional layers, the VGG comprises more weight layers, which, in addition to the depth of the network, can be seen as an advantage. Furthermore, the fact that the convolutional and pooling properties are the same for every layer simplifies its implementation.

GoogleNet

Szegedy *et al.* [121] proposed the GoogleNet architecture, a convolutional neural network that comprises 22 parameter layers and 5 pooling layers. This is based on the Inception architecture, employing Inception modules that allow the network to choose the size of the filters and the type of layers. Bearing this in mind, it is possible to achieve a total of 5 million parameters. 1×1 convolutions using 128 filters are also employed in this architecture in order to reduce dimensionality and rectified linear activation is considered. GoogleNet has seen several follow-up versions, being the Inception-V4 the most recent one [127].

ResNet

The ResNet architecture was introduced in 2015 by He *et al.* [122] representing a major breakthrough in terms of the concept of depth. The ResNet comprises 152 layers structured in residual blocks, which allow to overcome issues related to the training of very deep convolutional neural networks. Residual blocks (Figure 2.15) allow to skip some of the layers, working as shortcuts, in a process called "skip connection". Assuming that the underlying mapping corresponds to $H(x)$, where x represents the input, it is possible to establish a residual mapping where stacked nonlinear layers can be fitted, which is given by $F(x) = H(x) - x$. The original function can then be recast from $F(x) + x$. Therefore, for each residual block, there is a residual mapping that allow stacked layers to fit in, instead of directly fitting the desired underlying mapping. As a consequence, ResNet allows that new relevant information is extracted without using extra parameters [122].

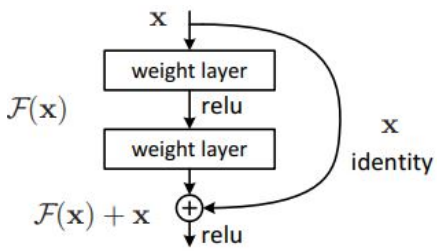
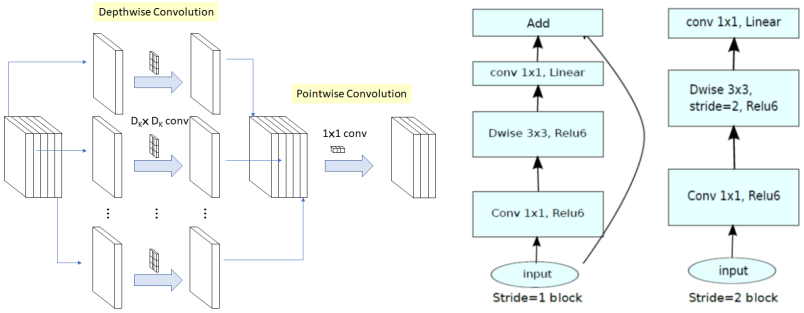


Figure 2.15: Possible residual block [122].

MobileNet

Howard *et al.* [128] proposed the MobileNet, a lightweight neural network architecture designed for mobile and embedded vision applications. The size and complexity of the model are reduced due to a depthwise separable convolution technique. This technique consists of a depthwise convolution in all channels followed by a pointwise convolution, which is a 1x1 convolution that allows to change the dimension of the output. After each convolution step, batch normalization and also a nonlinearity using a ReLU function are applied. It is worth noting that the MobileNet using depthwise separable convolution contains less 25.1 million parameters and only 1% loss in accuracy when compared to the MobileNet using a standard convolution.

Following the MobileNet, the MobileNetV2 network was proposed by Sandler *et al.* [129] in 2018. This network introduces a new module with an inverted residual structure where the input and the output of the residual block consist of thin bottleneck layers. Besides, in this network, the nonlinearities presented in the narrow layers were removed to preserve representational power. In Figure 2.17 are presented the architectures of both MobileNet (2.16a) and MobileNetV2 (2.16b).



(a) MobileNet [128]. (b) MobileNetV2 [129].

Figure 2.16: MobileNets acrchitectures.

EfficientNet

In 2019, Tan *et al.* [130] introduced the EfficientNet, a novel network architecture and scaling approach able to uniformly scale all dimensions of depth, width and resolution using a simple but effective compound coefficient, as illustrated in Figure 2.17. The authors developed a baseline network, the EfficientNet-B0, to demonstrate the effectiveness of the proposed scaling method in obtaining new scaled networks, namely from the EfficientNet-B1 to the B7. This network comprises inverted bottleneck residual blocks, as the ones used in MobileNetV2 [129], with a squeeze-and-excitation optimization [131].

2.5.3 Backpropagation

One of the methodologies used for training NN relies on the backpropagation concept. Backpropagation enables to compute an objective function's gradient, namely the loss function, taking into account the weights throughout the layers, and, hence, consisting of a direct application of the derivatives chain rule [109]. This process may be done in combination with a gradient descent

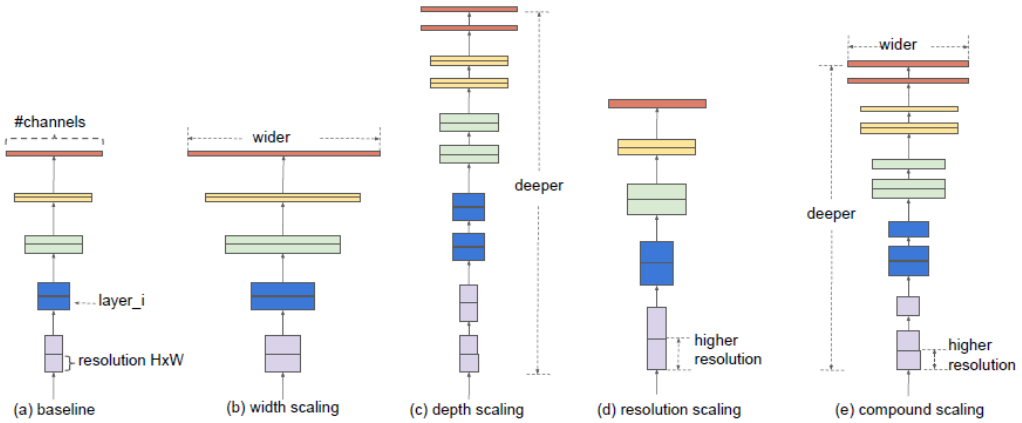


Figure 2.17: Model Scaling. (a) is a baseline network example; (b)-(d) are conventional scaling that only increases one dimension of network width, depth, or resolution. (e) is the compound scaling method that uniformly scales all three dimensions with a fixed ratio proposed in [130].

algorithm, as the Stochastic Gradient Descent (SGD), that intends to minimize the difference between the prediction and the ground truth. Therefore, the aim is to minimize the loss function, by updating the hyperparameters (weights and bias) in the opposite direction to the gradient [132].

2.5.4 Strategies for Model Performance Improvement

Training a model suitable for a certain task may be a demanding process. Many times, the problem requires the implementation of too complex models that are impossible to be properly trained with the amount of available data. As a consequence, some strategies, as transfer learning or data augmentation, have been developed in order to overcome this limitation.

Transfer Learning is a popular technique used in computer vision that intends to improve the algorithm's performance, leveraging knowledge from another task [133]. This approach takes advantage of the fact that the first CNN layers enable the extraction of more generic features that are shared among different datasets. Therefore, it is possible to transfer information from a model pre-trained on a sizable dataset, such as the ImageNet that contains around 1.4 million images divided by 1000 classes, and fine-tune the last layers (i.e. the more specific ones) to the desired task [134]. Hence, larger CNNs can be trained using a smaller amount of data, without the problem of overfitting.

Data Augmentation is another approach commonly employed when the amount of labelled data is insufficient for the problem purpose. It consists of the creation of synthetic data from the existing dataset, being quite useful in image-related tasks. The involved techniques may comprise not only rotations, scale transformations, or flippings, but also changes in color, amount of noise, and other more complex techniques.

2.6 Object Detection

Object detection is a fundamental and challenging problem of computer vision, having attracted the attention of the scientific community in recent years [135]. Its aim is not only to identify the categories of objects contained in an image but also to return their spatial location by means of a bounding box, for instance [136]. The classic approaches of object detection were based on the sliding-window paradigm, becoming computationally expensive [137]. As such, these techniques have been evolving, and deep learning approaches are being widely adopted in this field [138]. Deep learning-based object detection comprises two main categories of detectors: two-stage detectors and one-stage detectors [139]. Typically, two-stage detectors achieve higher localization and object recognition accuracy, whereas one-stage detectors are associated to higher speed.

2.6.1 Two-stage detectors

With two-stage detectors, the detection task is divided into two phases: the first one concerning the generation of proposals, and the second one that intends to make predictions regarding the proposals. Therefore, the detectors start by proposing the regions of interest (RoI) in the image, this is, the regions that eventually contain an object, intending that each object contained in the image belongs to at least one of the proposed regions. Then, a deep learning-based model is used, which, after feature extraction, is able to predict the category of the object in the proposed region [135].

R-CNN [140], proposed by Girshick *et al.*, is a classical example of this methodology. The system is composed of three modules concerning region proposal, feature extraction, and region classification, which are presented in Figure 2.18. Through selective search [141], 2000 regions of interest are generated for each image, being the approach designed to reject regions related to the background. Each proposal is warped or resized and it is used a pre-trained CNN to extract a fixed-length feature vector. Regions are then classified using class-specific linear SVMs and bounding-boxes around the objects are adjusted by linear regression. R-CNN presents some limitations, namely because training and testing are extremely time-consuming since the features are separately extracted from each region. Moreover, these features are stored on the disk, which requires extensive memory capacity while training. Therefore, He *et al.* proposed **SPP-net** [142] in order to decrease R-CNN training time and to make it learn more distinctive features. Differently from the R-CNN, instead of performing convolutions for each region proposal for feature extraction, SPP-net convolves the input image only once to obtain a feature map. The region proposals are then defined by correspondence between the original image coordinates and the feature map coordinates, and features are extracted by a Spatial Pyramid pooling (SPP) layer, avoiding the need to warp or resize regions.

Similarly to SPP-net, with **Fast R-CNN** [143] the entire image is processed by multiple convolutional and pooling layers in order to obtain a feature map. An RoI pooling layer, which is a particular case of the SPP layer, is then used to extract a fixed-length feature vector from the

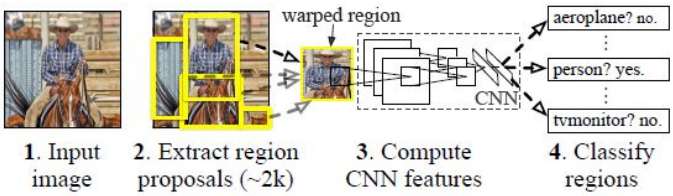


Figure 2.18: R-CNN system representation [140].

feature map, as presented in Figure 2.19. The features are provided into a sequence of FC layers resulting in two sibling output layers: one classification layer, responsible for generating the softmax probabilities of each object class, and one regression layer that outputs four real-valued numbers for each object that encode the bounding-box position. Fast R-CNN proved to be faster than R-CNN and SPP-net both in the training and test phases.

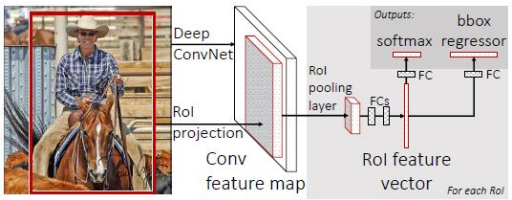


Figure 2.19: Fast R-CNN system representation [143].

Ren *et al.* proposed **Faster R-CNN** [144], which introduces a Region Proposal Network (RPN) that is used in combination with Fast R-CNN. This is a fully convolutional network that can be end-to-end trained in order to generate high-quality region proposals. This network enables to accelerate the generation of region proposals since it shares convolutional features of the whole image. It also introduces a method for different size object detection, using multi-scale anchors as reference. The following architecture is similar to the Fast R-CNN detector. Therefore, two outputs are provided, which concern object classification and bounding-box regression.

Mask R-CNN [145] was proposed by He *et al.* and is an extension of Faster R-CNN. The system considers an additional module to predict an object segmentation mask in a pixel-to-pixel manner on each RoI. It consists of a fully convolutional network that runs in parallel with the already existing branches of Faster R-CNN (classification and bounding-box regression). Nevertheless, some alterations are performed in Faster R-CNN. Instead of an RoI pooling layer, it is used a RoIAlign to extract a small feature map from each region, in order to prevent RoI and extracted features to be misaligned due to spatial quantization operations of RoI pooling. Mask R-CNN surpassed all existing single-model entries, including the COCO 2016 challenge winners, being a breakthrough in the object detection field.

2.6.2 One-stage detectors

One-stage detectors, on the other hand, are able to predict objects' bounding boxes, without the need of proposed regions of interest [135]. The most popular detectors of this category in-

clude YOLO [146], single shot detector - SSD [147], RetinaNet [137], and EfficientDet [148] approaches.

YOLO [146], You Only Look Once, was proposed by Redmon *et al.* and considers a single convolutional network to predict both bounding boxes and class probabilities. YOLO, firstly, divides the image into a grid, assigning the object to the cell that contains its mid-point. It is calculated a confidence score for the bounding boxes predicted by each cell, taking into account the probability of a box contains an object and the accuracy of an object is contained in that cell, which is given by the intersection over union. Thereby, although YOLO shows to run extremely fast (it is able to detect 45 frames per second), its location accuracy falls short of the two-stage detectors. Some improved versions of YOLO were already developed, namely YOLOv2 [149] and YOLOv3 [150].

Concerning **SSD** [147], this system also splits the image into grid cells and assign default bounding boxes of different dimensions. For each default bounding box, a prediction of the offsets and confidences concerning each object class is made. SSD considers several feature maps extracted with VGG-16 to deal with multiple object dimensions. Depending on the receptive field of each feature map, it is possible to detect objects of different scales. Therefore, the final prediction results from a combination of all feature maps detections. As the number of predictions is quite higher than the number of objects, hard negative mining is considered, in order to decrease the number of negative proposals and prevent class imbalance. SSD proved to be effective in object detection, outperforming Faster R-CNN in terms of speed and localization accuracy.

Lin *et al.* [137] developed **RetinaNet** an object detector that uses as classification loss function the focal loss, instead of the standard cross-entropy loss. This function was proposed with the aim of reducing the number of negative locations which are typical of one-stage detectors. Therefore, the focal loss enables to decrease the weight of the loss that is assigned to the well-classified or easy samples, focusing the model on the hard samples. The detector is composed of a main network that computes the convolutional feature map of the entire input image, and two subnetworks responsible for the object classification and the bounding-box regression, respectively. RetinaNet proved to be as fast as the previously described one-stage detectors, with the advantage of overcoming the class imbalance problem.

EfficientDet [148] was introduced by Tan *et al.*, presenting a new family of highly efficient, accurate and faster detectors. These detectors use EfficientNets [130] as backbone networks along with Bidirectional Feature Pyramid Networks (BiFPNs) that allow fast multi-scale feature fusion. Since the contribution of features with different resolutions to the final output is unequal, the authors proposed three methodologies that allow to assign weights to different feature maps. Moreover, inspired by the EfficientNets [130], the authors proposed a compound scaling method for the detectors that allows to scale the resolution, depth, and width of the backbone, feature network, and box/class prediction networks at the same time. Comparing to the other state-of-the-art object-detectors, EfficientDet proved to be the most efficient one. In Figure 2.20, it is possible to find the architecture of this detector.

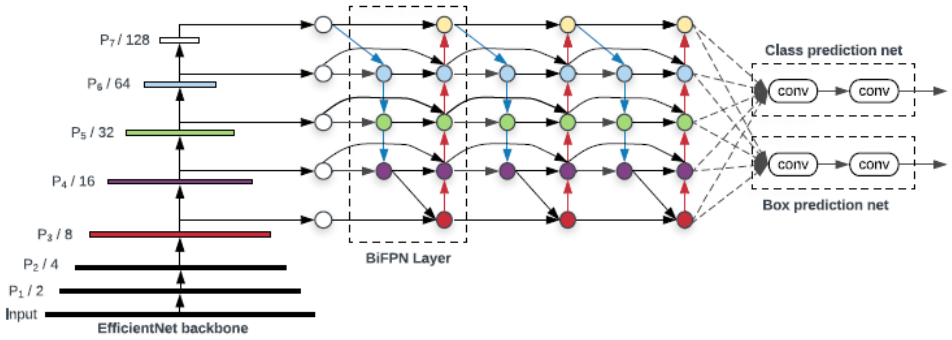


Figure 2.20: EfficientDet architecture [148].

2.7 Incremental Learning

As previously mentioned, data is continuously evolving and being collected whereby the access to all the information is rarely achieved at once [22]. Thus, it is possible that over time, the acquisition properties of data change, leading to alterations that affect its distribution. Nevertheless, traditional ML-based computer vision systems are static, requiring a dataset with fixed data distribution in order to optimize the learning process. Therefore, in order to complement their learning with new information from unseen data, these models need to be retrained using both previous and new data, which may be unfeasible due to the high computational cost involved or because data from the past may not be available anymore due to memory issues, for instance [151]. The capability of the models to adapt to new conditions is then closely related to their capability of learning incrementally. Taking this into account, there has been an increased interest in models able to be incrementally modified as new data becomes available, without compromising the knowledge acquired in previous tasks, nor the need of having access to all data at the same time [152].

Incremental learning, also known as continuous learning or lifelong learning [153], intends to overcome these situations, comprising models able to preserve and extend the already acquired knowledge in order to solve new tasks [154]. However, it is a demanding process to effectively learn without resorting to future data (and possibly to previous data) nor forgetting the already learned information. For these reasons, learning new tasks sequentially remains challenging and incremental learning approaches typically have to face several issues such as concept-drift [155], catastrophic forgetting [156], or the stability-plasticity dilemma [157][158], being this last one also reported for biological systems [159][160].

2.7.1 Challenges Addressed by Incremental Learning

Regarding the **concept-drift**, this incremental learning challenge is essentially associated to variations in the data stream over the various tasks, being possible to identify two different scenarios: virtual or real concept-drift. Virtual concept-drift occurs when the input data distribution changes, due to the imbalance of classes over time, for instance. Real concept-drift, on the other hand, concerns novelty on data, which may result from the insertion of new features or from different

image acquisition protocols, or even the introduction of new classes, leading to alterations in the capacity of the model to solve the new problem [155][160].

Another major challenge (and probably the most popular one) when implementing incremental learning approaches relies on the **catastrophic forgetting** problem [156]. This problem occurs when a model is sequentially trained on new concepts and a performance degradation is verified on the previously learned concepts as new data is added [161][162][163]. Thus, catastrophic forgetting prevents models to learn multiple tasks continuously, and for that reason, strategies able to mitigate this problem are being increasingly developed [164][165].

Moreover, due to memory limitations, sometimes it is not possible to store all data related to previous tasks when new tasks are added [166]. Incremental learning approaches then differ in the way they handle memories of the previously learned tasks, in order to avoid the catastrophic forgetting. Different mechanisms may be used to store this information, such as raw data, model weights, regularization matrices, and others. However, it is almost impossible to predict which information will be important in the future, and, for this reason, a trade-off between the precision of the stored information and the acceptable forgetting should exist. This trade-off is also called the **stability-plasticity dilemma**, which is related to the required balance between the learning stability and plasticity of the model. Stability concerns the model ability to preserve previous knowledge and plasticity denotes the adaptation to new knowledge. On the one hand, if a model is too stable, although it preserves the acquired knowledge, it cannot learn new information. On the other hand, if a model is too plastic, it easily accommodates new knowledge, but prior knowledge is forgotten [167][168].

2.7.2 Content Update Scenarios

Describing the context of an incremental learning problem is essential to understand how it can be approached. The complexity of incremental learning models may depend on the type of content that is used to update them at each training batch, and therefore, three different scenarios can be considered [155][169]:

• New Instances - this scenario assumes that the new training batches are composed of data belonging to the same classes contained in the previous batches, but which may include new information to be learned.

• New Classes - on the other hand, it is also possible to admit that the content of the new training batches includes classes that were never observed in previous batches, as presented in Figure 2.21.

• New Instances and Classes - this is a more realistic scenario which considers that new data is composed by both new instances (examples) of previously observed classes and new classes.

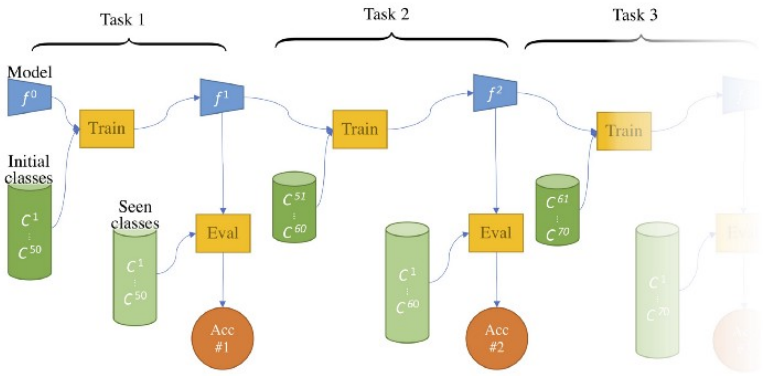


Figure 2.21: Incremental learning process for new classes update [170].

2.7.3 Incremental Learning Strategies

In order to alleviate the catastrophic forgetting when training models incrementally and while dealing with the concept-drift, different techniques have been taken into account. Due to the sudden and increased motivation for incremental learning in the last years, there is still no well-defined and common terminology among the scientific community in this field. However, having as reference Lesort *et al.* [155], Kember *et al.* [167], and Zenke *et al.* [171], besides the baseline strategies, these methodologies may be associated to a three-way categorization that comprises architectural strategies, regularization strategies, and rehearsal strategies. It is worth noting that these strategies do not contradict each other and can be used in combination to mitigate the catastrophic forgetting.

2.7.3.1 Baseline strategies

Before moving on to the more complex approaches, it is possible to define two methodologies that can be used as standard baselines for incremental learning approaches, which are the *Naïve* and the *Cumulative* (also called *Offline*) strategies.

• Naïve - In this strategy, the model is simply fine-tuned with the new examples without using any mechanism to avoid the catastrophic forgetting [172]. Thereby, if the data distribution faced by the model varies a lot, this approach is specially prone to lead to a decrease on the previously acquired knowledge (i.e. catastrophic forgetting).

• Cumulative - This strategy intends to tackle the catastrophic forgetting by training the model using all previous examples together with the new ones. Therefore, since it requires all data to be stored in memory, it cannot properly be defined as an incremental learning approach, but rather as a means of comparison [173].

2.7.3.2 Architectural strategies

Since the architecture of a model strongly influences the way it learns, the catastrophic forgetting problem may be smoothed with this kind of strategies by resorting to network architecture manipulations, without changing the objective function. Thus, these changes can be seen by the accommodation of new neurons or layers, changing the activation functions, or even by freezing specific weights within the network.

2.7.3.3 Regularization strategies

Regularization strategies, on the other hand, extend the loss function with a regularization term, constraining the update of the weights depending on the neurons importance. Therefore, it is possible to improve the stability of the model, since the previously learned and important knowledge is not affected, which, consequently, allows to mitigate catastrophic forgetting. Regularization strategies further comprise two categories: weight regularization strategies and distillation strategies.

• Weight Regularization Strategies:

In this type of strategies, a regularization term is added to the loss function of the new task, which can be obtained by:

$$L(\theta) = L_n(\theta) + \lambda R(\theta_i) \tag{2.5}$$

where L_n represents the loss function of the new data, λ consists of an hyperparameter, R is the regularization term, and θ_i is related with the relevant parameters to the old knowledge.

Thus, the update of the most important weights can be limited and, thereby, the knowledge learned in the previous tasks can be preserved.

• Distillation Strategies:

With distillation strategies it is intended that after a larger neural network learns a certain task, a smaller neural network will mimic its ability. Hence, when the two networks are given the same input, it is supposed that both of them generate the same output. The new model is then coherent with the old one, since both the new and the old weights are constrained. Knowledge distillation [174] is implicit in this kind of strategies, minimizing the loss when knowledge is transferred between models. In this case, the probability of each class can be addressed using a softmax output layer, which is obtained by:

$$q_i = \frac{exp(z_i/T)}{\sum_j exp(z_j/T)} \tag{2.6}$$

where q_i concerns class probability, z_i corresponds to the logits, and T is a temperature coefficient that when set to 1, corresponds to the standard softmax function.

2.7.3.4 Rehearsal strategies

The approach considered by the rehearsal methods to mitigate catastrophic forgetting consists of reproducing a subset of previous data on the current model, in order to strengthen connections regarding information that was already learned, as observed in Figure 2.22.

Pseudo-rehearsal is another possible incremental learning strategy that involves the generation of pseudo patterns based on the input data distribution which are used by the new model, allowing to stabilize older memories without storing all previously learned information.

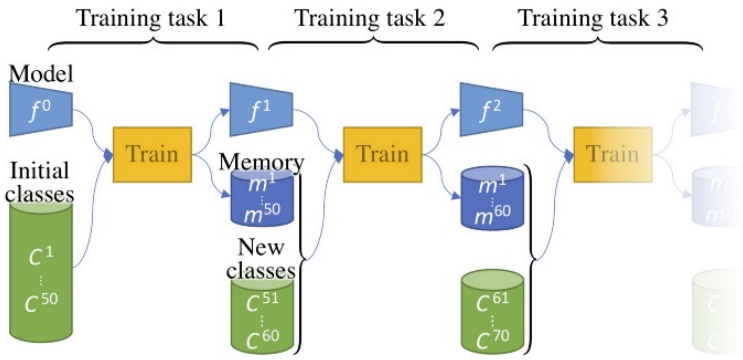


Figure 2.22: Rehearsal strategy for new classes update [170].

In Chapter 3, different methodologies concerning the three presented types of incremental learning strategies that have been developed in the last years in order to mitigate catastrophic forgetting are reviewed and compared.

2.8 Performance Metrics

Several key metrics can be used to evaluate the performance of the developed computer vision models. The selection of the most suitable metrics must take into account the intended goals of the work.

Although the confusion matrix, represented in Figure 2.23, is not truly a performance metric, it enables a representation of the true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN).

- TP - when both the prediction and the ground-truth are positive
- TN - when both the prediction and the ground-truth are negative
- FP - when the prediction is positive but the ground-truth is negative
- FN - when the prediction is negative but the ground-truth is positive

These concepts are fundamental for the understanding of other metrics, such as accuracy, precision, sensitivity, specificity, and others. In Table 2.5, some of the most common performance metrics are explained .

Other metrics as the Receiver Operating Characteristic (ROC) curve or the Area Under the Curve (AUC) are also commonly used to evaluate the models.

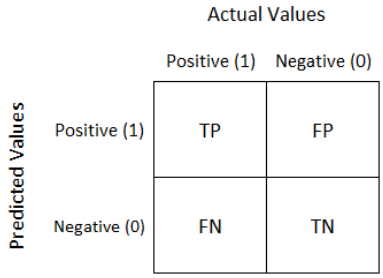


Figure 2.23: Confusion matrix representation.

ROC curve is obtained by plotting the TPR against the FPR (1-Specificity) for several threshold properties and enables to visualize the performance of a classification model for each one of them, as represented in Figure 2.24. The optimal cut-point value corresponds to the trade-off between sensitivity and specificity that provides the highest sensitivity and the highest specificity. Hence, the further away from the diagonal line (random guessing) the ROC curve is, the better the model's performance. This leads us to the concept of AUC, which corresponds to the area under the ROC curve. Therefore, the AUC gives the probability of the classifier to rank a randomly chosen positive sample higher than a randomly chosen negative one, varying between 0 and 1 [175].

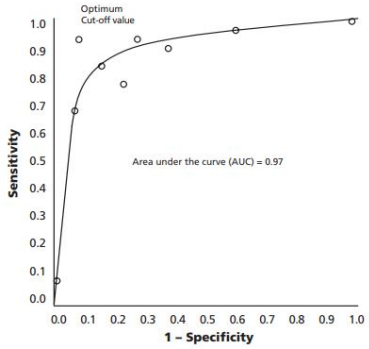


Figure 2.24: Receiver Operating Characteristic (ROC) curve [176].

With respect to the evaluation of **incremental learning**, different metrics have been developed in order to assess the performance of the corresponding strategies.

For instance, Lopez *et al.* [177] proposed the Backward Transfer (BWT), and the Forward Transfer (FWT) in order to evaluate the capacity of the model to transfer knowledge across different tasks. The higher the BWT and the FWT, the better is the model's performance.

• *Backward Transfer* (BWT) represents the influence of the current task on the performance of the preceding task. If the task leads to an increase on the performance of the previous task, a positive Backward Transfer is verified. On the other hand, if learning a new task results in a decrease on the preceding task's performance, we are facing a situation of negative Backward

Table 2.5: Performance metrics.

Metric	**Formula**	**Definition / Main applications**
Accuracy	$\frac{TP+TN}{TP+FP+TN+FN}$	Number of correctly classified samples. Applications: Classification, Object Detection, Segmentation
Precision	$\frac{TP}{TP+FP}$	Ratio of positive labeled samples that actually are. Applications: Classification, Segmentation
Sensitivity, Recall or True Positive Rate (TPR)	$\frac{TP}{TP+FN}$	Proportion of correctly identified positive samples. Applications: Classification, Segmentation
Specificity	$\frac{TN}{TN+FP}$	Proportion of correctly identified negative samples. Applications: Classification, Segmentation
False Positive Rate (FPR)	$\frac{FP}{TN+FP}$	Ratio between the number of FP and the number of negative samples. Applications: Classification, Segmentation
F1-Score	$2x\frac{PrecisionxRecall}{PrecisionxRecall}$	Harmonic mean of precision and recall. Applications: Classification, Segmentation
Jaccard coefficient	$\frac{TP}{TP+FP+FN}$	Measure of the similarity between the ground truth and the predicted labels. Applications: Segmentation
Dice coefficient	$\frac{2TP}{2TP+FP+FN}$	Measure of the spatial overlap extension between two binary images. Applications: Segmentation
Intersection over Union (IoU)	$\frac{target \cap prediction}{target \cup prediction}$	Measure of the overlap between two regions. Applications: Object Detection, Segmentation

Transfer. Hence, the computation of the BWT is given by:

$$BWT = \frac{1}{T-1}\sum_{i=1}^{T-1} R_{T,i} - R_{i,i} \tag{2.7}$$

where T represents the tasks and R is the training-test accuracy matrix, where $R_{i,j}$ is the test classification accuracy of the model on task t_j after observing the last sample from task t_i.

• *Forward Transfer* (FWT) consists of the opposite of BWT, representing the influence that the current task has on the performance of a future task. The FWT is then calculated by:

$$FWT = \frac{1}{T-1}\sum_{i=2}^{T-1} R_{i-1,i} - b_i \tag{2.8}$$

where b corresponds to the test accuracies vector for each task at random initialization.

Nevertheless, although several other metrics have been proposed in the last years in order to assess the accuracy, train/test time, or storage requirements of the incremental models [167][178][179], there is still no consensus among the computer vision community in evaluating incremental learning algorithms and, the equations of BWT and FWT have even already suffered some modifications by other authors [178].

Chapter 3

Literature Review

3.1 Medical Imaging Modalities Classification

The main goal of medical imaging modality classification is to distinguish different types of medical images, such as CT, MRI, X-ray, and others, as presented in Section 2.3. This objective follows the need of having effective classification systems that allow to better organize medical records, which every year suffer a huge increase in the amount and diversity of saved medical information.

Taking this into account, it is pressing to develop methods able to facilitate access to this data, which can be a demanding task. Medical image modality may be one of the most important filters for image retrieval, allowing to restrict the results of the search, and get more accurate outcomes. Thus, many studies have been done in this sense, in order to achieve automatic algorithms capable of classifying medical imaging modalities. These systems comprise both feature engineering methods and deep learning-based approaches.

Therefore, the developed works using these two main approaches for modality classification will be presented in this Section. Moreover, an overview of the considered methodologies can be found in Appendix A.

3.1.1 Hand-crafted feature based approaches

Kalpathy *et al.* [83] proposed a hierarchical neural network based classifier able to classify grey-scale images as CT-scans, MRI or X-ray, as well as colour images as microscopic or endoscopic images. For this, the authors used the CISMeF database and the ImageCLEFmed 2006 database, respectively, considering six modalities of each one. As inputs to the network a combination of histogram and texture features was taken into account. Two different models were considered in the developed system, depending on the images' characteristics. In the case of colour images, the neural network implemented a two-layer architecture, whereas for grey-scale images it was used a multilayer perceptron architecture with one hidden-layer. The results show an accuracy greater than 95% for both classifiers, being higher in the case of colour images.

The work developed by Song *et al.* [180] as part of their participation in the ImageCLEF 2012 consisted of a mono-modal visual-based image classifier. The experiments were based on

three kinds of feature extraction techniques: Edge Histogram, Tamura and Gabor. The authors made different combinations of these features and used the LibSVM library to train one-versus-all classifiers. The better performance was verified when the three features were considered at the same time, achieving an accuracy of 60.01% on image modality classification.

Khachane *et al.* [181] used support-vector machine (SVM) and k-nearest neighbors (k-NN) supervised techniques and a fuzzy rule-based approach for image modality classification. The experiments were performed on five different image modalities, comprising CT, X-Ray, MRI, Microscopic, and Ultrasound images. The extracted features were the mean, standard deviation and contrast of the image. The fuzzy systems were also included in the training phase in order to design the member functions and to plan the rule base. The authors defined a set of twenty-three rules to classify each medical image in one of the mentioned modalities. The developed model reached an accuracy of 86%, which is worse than the obtained results using the SVM classifier (88%), but higher when compared to k-NN (84%).

A model implementing discrete Bayesian network classifiers (BNC) for hierarchical medical image modality classification was proposed by Arias *et al.* [182], using the ImageCLEFmed 2013 dataset. From a combination of five initial sets of descriptors (Bag-of-Visual-Words (BoVW)[183] using Scale Invariant Feature Transform (SIFT), Bag-of-Colors (BoC), Color and Edge Directivity Descriptor (CEDD), Fuzzy Color and Texture Histogram (FCTH) and Fuzzy Color Histogram (FCH)), the authors were able to create 31 different combinations that were used for data pre-processing, as well as for feature subset selection. For the classification problem, various semi-naive BNC models were analysed, being the averaged one dependence estimator (AODE) the one that provided better results when an hierarchical classification was conducted. The highest accuracy obtained by this model was 69.21%, corresponding to the 3^{rd} place of ImageCLEF 2013 contest for the modality classification task.

Cao *et al.* [184] adopted two strategies to deal with the medical modality classification task of ImageCLEF 2012. Firstly, the authors tried to augment the training examples, using other databases and images beyond the original ImageCLEF2012 dataset, and, then, explored several feature extraction approaches. Seven sets of features based on Local Binary Patterns (LBP), edge and color histograms, SIFT descriptors, and others were created and used for modeling. A multiclass SVM classification was taken into account, by means of the LibSVM library and a 5-fold and a 3-fold cross validation were performed in the original dataset and the augmented one, respectively. Early, late, and kernel fusion were analysed as feature fusion methods. The best performance was achieved when considering the augmented dataset in combination with the early fusion strategy, corresponding to an accuracy of 69.7%.

The strategy adopted by Valavanis *et al.* [185] consisted of a merge between BoVW and BoC models, applying early and late fusion. These models contain Pyramid Histogram of Visual Words (PHOW) and Quad-Tree Bag-of-Colors (QBoC) descriptors, respectively, that were considered for image visual representation. The classification task involved the implementation of linear SVMs. The best result achieved, using only visual categorization, corresponded to an accuracy of 84.01% associated to an early feature fusion.

The model proposed by Markonis *et al.* [186] consisted of a visual and textual approach that uses a k-NN classifier for medical image modality classification and retrieval. The authors used a Bag–of–Visual–Words approach considering SIFT descriptors for each image. Through a clustering method, it was possible to define the visual words, which corresponded to the centroids of the descriptors clusters. A GNU Image-Finding Tool (GIFT) mechanism was also considered in the classification task, allowing to obtain the distance value for the nearest neighbor classification (1-NN). The best accuracy was obtained using a mix of textual and visual characteristics (86.9%), being the results of GIFT using 1-NN classification the worst.

Also Dimitrovski *et al.* [82] developed a model for medical image modality classification as part of the ImageCLEF competition. Beyond four visual and one textual types of features, the authors also used combinations of them to describe the images. Visual features comprised LBP, FCTH, CEDD, and SIFT descriptors. Regarding the textual feature, it consisted of a standard Bag-of-Words (BoW) textual representation associated to TF-IDF (term frequency/inverse document frequency) weighting. Similarly to Cao *et al.* [184], the classification task also involved the implementation of SVM through the LibSVM library. Different SVM classifiers were used for visual and textual features, due to the variations verified among them. It was considered a one-vs-all strategy using a binary classifier for the multi-class classification. Since this approach led to unbalanced data, the weights of positive and negative classes were adjusted. The performance of the algorithm was evaluated in ImageCLEF 2011, 2012 and 2013 databases. The best accuracy was verified for the 2011 database when visual and textual features were mixed, corresponding to a value of 87.10%.

Within the scope of ImageCLEF 2013 contest, also Kitanovski *et al.* [187] implemented a method for medical image modality classification, similar to the one described in [82] since some authors are common to both works. In the proposed model each SVM classifier was trained with χ^2 kernel and an one-vs-all approach was used. As features, the authors considered SIFT descriptors, using a BoVW approach, and TF-IDF weights of the surrounding text. These two features were combined using high level feature fusion strategy. Due to the imbalanced ratio between classes, the weights were adjusted. The highest accuracy was verified for the mixed run, this is, when the high level feature fusion was performed: it was achieved an accuracy of 77% using the ImageCLEF 2012 dataset and of 78.04% using the ImageCLEF 2013 dataset.

Besides the work presented by Pelka *et al.* within the scope of ImageCLEF 2015 competition[188], the authors developed another model that combines both visual and textual features for the image modality classification purpose [189]. As visual features, local and global features were taken into account. Regarding local features, the authors considered Bag-of-Keypoints (BoK), using SIFT descriptors, and Pyramid Histogram of Oriented Gradients, whereas global features comprise Tamura, Gabor, FCH, and others. Regarding textual features, a Bag-of-Words approach was applied. For the classification task, a Principal Component Analysis (PCA) was used in order to reduce the dimensionality of features. Concerning modality classification, beyond a Random Forest classifier, a multi-class SVM was also considered. The final prediction consisted of a fusion between Random Forest and SVM prediction. The best accuracy achieved by the model was of

67.6%, corresponding to a mix between visual and textual features.

The approach developed by Wu *et al.* [190] was similar to the one described in [187]. The authors used visual features as SIFT, LBP, Gabor, and Tamura, in combination with textual features, defined by a vector space model (VSM) weightened by TF-IDF, through a l_p-norm multiple kernel learning (MKL). For the multi-class classification task, it was also considered an one-vs-all approach, using an SVM classifier. Comparing to other feature combination methods as early or late fusion, MKL provide better results, corresponding to an accuracy of 95.15%.

Gál *et al.* [191] also proposed a model that combines visual and textual features, using normalised kernel function in SVM for medical image modality classification. The extracted features comprised the caption text, the colour histogram, the mean of pixels, meta-data, Bag-of-Visual-Words, considering SIFT descriptor and TF-IDF weighting, and others. The authors used different combinations of the mentioned features. The best performance corresponded to an accuracy of 86.03% in the test phase.

In the work developed by Csurka *et al.* [81] also visual and textual features were considered for modality classification. For image representation, the authors used both BoVW and its Fisher extension, also known as Fisher Vector. The extracted features comprised SIFT-like Orientation Histograms and local color averages and standard deviations of the RGB channels, that were combined using late fusion. Regarding textual representation, BoW was employed. A logistic regression classifier with Laplace Prior was used for each modality, in order to achieve the images' classes. Considering only visual features, the best performance was verified when the features were fused and the image was represented by the Fisher vector, corresponding to an accuracy of 86.9%. However, when the textual features were also considered, this value increased, achieving categorization accuracy of 94.4%.

That being said, although hand-crafted designed approaches for medical image modality classification are prevalent in the literature, it can be seen that there is a great variation between the obtained results. This occurs because the algorithm's efficiency is highly dependent on the features' selection, both in terms of the number and type of the extracted features. For this reason and also because the considered methodologies are typically specific for a certain task or dataset, involving different procedures to be used whenever a new task is required, it is a demanding work to predict the most suitable features for modality classification. As a result, much of the effort in this problem concerns the design of a strategy for image pre-processing and data transformation, which can be an intensive and time-consuming work.

Hence, less human-dependent approaches are increasingly being developed and adopted, in order to achieve more efficient medical image modality classifiers. Thereby, some state-of-the-art examples concerning this other kind of methodologies are explored in the next Section.

3.1.2 Deep neural network based approaches

CNNs are able to learn features autonomously from data and, for this reason, are preferable approaches in many situations. Regarding modality classification, several techniques using deep

learning-based models can be found in the literature, some of them using transfer learning techniques for feature extraction.

Starting by the work developed by Kumar *et al.* [192], the authors considered a method that uses a combination of different convolution neural networks (CNN) architectures for medical image classification, extracting features at various levels. AlexNet and GoogleNet architectures were fine-tuned, being then used as image features extractors and as classifiers. The classification comprised both softmax and three one-vs-one multiclass SVMs classifiers considering the features extracted with AlexNet, GoogleNet and both, respectively. The image class was determined using the probabilities from the combination of softmax and SVMs classifiers. The method achieved competitive accuracy values (82.48% - top 1 and 96.59% - top 5) comparing to other baseline methods, showing its ability to differentiate between image modalities with subtle variations.

Within the scope of ImageCLEF 2016 competition, another methodology for medical imaging modality classification was proposed by Kumar *et al.* [193]. In this work, the AlexNet architecture was pre-trained on the ImageNet dataset and fine-tuned using medical images of the ImageCLEF 2016 database, that allowed to extract features more specific to the problem context. Besides data augmentation, also dropout was considered in order to avoid overfitting and PCA was used to reduce the dimensionality of the feature vector. For the classification problem, six crops of each image were taken into account, and a multi-class support vector machine was employed, allowing to obtain the posterior probability of each class. The best result of this approach corresponded to a correctness of 77.55%, associated to the mean SVM probabilities of all crops.

Also Semedo *et al.* [194] participated in ImageCLEF 2016 competition suggesting an approach for modality classification. In this work, different CNN models were considered, namely two VGG-like models, which differ in the depth, and one model based on using Parametric ReLU (PReLU) activation function in the hidden layers with Batch Normalization. Regarding the VGG models, a ReLU function was used in all of the hidden layers. A softmax function was employed in the last layer of the three models, assigning the classification of each image. The best performance corresponds to an accuracy of 65.31%, resulting from the deeper VGG model and considering the dropout technique to avoid overfitting.

Yu *et al.* [19] started by proposing a method that trains from scratch various CNNs for medical images modalities classification. The authors considered a DropConnect (DC) technique (a generalization of Dropout)[195], in order to avoid the overfitting problem due to the small amount of data. After training the created DNNs, these were combined using different fusion strategies (average, maximum, majority and median) to improve the algorithm's performance. In this work, the ImageCLEFmed 2013 dataset was taken into account. The best results were achieved by training 5 neural networks and combine them using a fusion strategy based on average. Also, the input images were resized to 192x192 pixels, achieving an accuracy of 74.90%.

The subsequent work developed by Yu *et al.* [196] consisted of a transfer learning approach using two data augmentation strategies for medical image modality classification. The authors started by consider VGGNet and ResNet networks with different depths and pre-trained on ImageNet natural image dataset [126]. The first layers, this is the ones corresponding to the most

generic features, were fixed, and only the layers corresponding to the domain-specific features were trained. After this, a CNN with only six weight layers was trained from scratch. The prediction probabilities for each CNN were obtained using the softmax function in the last layer. The results of the three CNNs were then combined in order to obtain the image class. In these experiments ImageCLEF2015 and ImageCLEF2016 datasets were considered, resizing the input images to 224x224 pixels. The developed CNN-6 achieved promising results, satisfying accuracy values of 66.13% and 81.86% for ImageCLEF2015 and ImageCLEF2016, respectively.

To implement a medical image modality classification system, also Hassan *et al.* [79] considered a transfer learning approach using a pre-trained ResNet50 for feature extraction. ImageCLEF2012 dataset was used and the extraction of features was followed by a linear discriminant analysis (LDA) classification, in order to ensure the maximum class separability. The proposed algorithm achieved an accuracy of 88%, being its performance also compared with hand-crafted features based classification approaches and using a softmax classifier instead of the LDA. In both cases, the developed model showed an improved classification accuracy.

Singh *et al.* [78] compared the results of seven different pre-trained DCNN (VGG-16, VGG-19, ResNet-50, MobileNet, Inception-v3, Inception-ResNet-v2, and Xception), using transfer learning for medical image modality classification. Being so, the weights regarding generic features were loaded, while only the last fully-connected layers were trained from scratch for feature extraction. A Logistic Regression classifier was then trained using the extracted features. The best performance was achieved by Google Inception-v3 model, corresponding to an overall precision of 99% on the test set, whereas the worst results were verified for VGG-16 and ResNet-50 models.

Khan *et al.* [77] intended to compare Deep Learned and Hand Crafted Features in the classification of medical imaging modalities. Regarding the proposed deep learning approach, the authors trained a CNN from scratch, that was composed by two convolutional layers, two pooling layers and a softmax layer as a classifier. On the other hand, the selected handcrafted method comprised a BoVW using SIFT keypoint descriptor in combination with Harris corner and LBP texture feature. It was verified that handcrafted features outperformed deep learned features, corresponding to an accuracy of 90.1% and 81.2%, respectively. Nevertheless, these results can be explained by the small amount of images used to train the CNN.

The system developed by Zhang *et al.* [197] used a synergic deep learning (SDL) approach. This methodology allows to enhance the ability of deep neural networks in differentiating images that may be confused, which is the case of brain and pleural CT images, for example, that seem to belong to different modalities. It consists of a data pair input layer and a pre-trained dual deep convolutional neural network in combination with a synergic signal system. This system allows to supervise the learning, verifying if the two images in the input belong to the same modality. The proposed SDL model was evaluated using the ImageCLEF2016 dataset, being its performance then compared with other state-of-the-art methodologies using the same dataset. The proposed algorithm achieved the highest accuracy, corresponding to 86.58%.

That being said, CNNs proved to be effective in the medical image modality task. It was verified that the usage of transfer learning allows to increase the performance of the created algorithms and that the combination of different models also provides promising results. Moreover, some of the presented works emphasized that deep learning models require large amount of data to be trained with the possibility of overfitting when considering small datasets. Hence, also data augmentation, or dropout techniques are common solutions to take into account in order to overcome this issue.

3.2 Skin Lesion Classification using Convolutional Neural Networks

Due to the similarities between different types of skin lesions, algorithms that aid dermatologists in the diagnosis task are an asset. In the last years, many studies considering CNNs have been done in this sense, allowing to assist the differentiation between benign lesions from those that may reach worrying malignant stages.

CNNs can either be employed as feature extractors being the classification performed by another classifier (SVM, k-NN,...) or directly as classifiers, in an end-to-end learning. In this last case, the network can either be trained from scratch or employ transfer learning. Therefore, in this section, some of the proposed approaches using CNNs for skin lesion classification will be explored.

3.2.1 CNN as feature extractor

Pomponiu *et al.* [198] presented a model that uses a pre-trained deep neural network to extract features that are useful for skin lesion diagnosis. As the available amount of data was reduced and there was a large diversity in images, the authors considered a transfer learning approach, using the architecture presented in [111]. Firstly, data augmentation was performed in order to avoid overfitting, as well as pre-processing which consisted of data resize and normalization. The features were extracted from the last three layers of the CNN and used by a k-NN classifier, which allowed to classify lesions. For the evaluation of the algorithm only cross-validation was performed, not being used a different test dataset. The proposed method achieved an accuracy higher than 90% for the features extracted from the last three layers of the CNN. Its performance was also compared with state-of-the-art hand-crafted feature models, proving to obtain better results.

The method proposed by Codella *et al.* [199] consisted of a combination of deep learning, sparse coding, and support vector machine learning algorithms for skin lesions classification. Hence, besides the features extracted from a pre-trained CNN, the authors also considered unsupervised feature learning, by means of sparse coding. For the classification task, dermatoscopic images from the International Skin Imaging Collaboration (ISIC) database were used, intending to classify not only melanoma vs. non-melanoma lesions, but also melanoma vs. atypical lesions. In order to achieve this goal, non-linear SVMs classifiers were employed and the obtained models were combined in late fusion. In the first classification task, an accuracy of 93.1% was achieved, whereas in the second one the authors reported an accuracy of 73.9%. Besides, a comparison

with prior state-of-art modeling approaches using only hand-crafted low-level features was made, having the proposed model achieved promising results.

For feature extraction, Kawahara *et al.* [200] slightly changed the pre-trained AlexNet architecture, converting fully connected layers into convolutional layers, and thus creating a fully-convolutional neural network. The responses of the new convolutional layer were then used to train a logistic regression classifier in order to classify several categories of skin lesions. The approach was validated on Dermofit Image Library, which contains 10 different lesion categories. The classification was made considering not only groups of 2 or 5 classes of lesions, but also over the full 10-class dataset, achieving an accuracy of 94.8%, 85.8%, and 81.8%, respectively, and outperforming previously presented studies.

3.2.2 End-to-end learning

The work developed by Esteva *et al.* [15] was a milestone in the field of skin lesions classification. For the first time, a dataset containing around 130 thousand clinical images of different diseases was used, a considerably higher amount of data than the previously analyzed. In this work, to train the Deep Neural Network (DNN), the authors considered a GoogleNet Inception v3 CNN architecture, pretrained on ImageNet[201], in order to improve the learning process. The classification into skin lesions was possible due to the fine-tuning of the CNN model using transfer learning. For data balancing, a novel tree-structured taxonomy for the skin diseases was also introduced, corresponding the individual diseases to the leaf nodes. In this approach, a binary classification was carried out, taking into account two critical use cases: keratinocyte carcinomas versus benign seborrheic keratoses, which as mentioned in Chapter 2 intends to represent the most frequent skin cancers; and malignant melanomas versus benign nevi, which intends to identify the deadliest types of skin cancer. The algorithm performance was then tested against 21 board-certified dermatologists on biopsy-proven clinical images. It was verified that the developed model outperforms more than half of the dermatologists at skin cancer classification. When tested on a larger dataset, the CNN evinced reliable results for the classification of skin lesions, presenting an AUC of 0.96 both for carcinoma and melanoma and 0.94 for melanomas classified through exclusively dermatoscopic images.

Similarly to Esteva *et al.* [15], also Haenssle *et al.* [17] considered an adapted pretrained GoogleNet Inception architecture for skin lesion classification, using transfer learning. The purpose of this study was to classify only melanocytic lesions through dermoscopic images and benign nevi. The performance of the developed model was then compared to the evaluation of 58 dermatologists, which was made in two levels. In level-I, clinicians were only provided with dermoscopic images, while in level-II, beyond these images, clinicians also had access to close-up images and additional clinical information. The results showed an improvement in the diagnostic performance of dermatologists when they received supplementary information (level-II). Regarding the performance of the algorithm, the CNN model achieved an AUC of 0.86 both for study level-I and level-II. On the other hand, the evaluation of dermatologists' performance took into account the values of the mean ROC area for the two situations, corresponding to 0.79 and 0.82,

respectively. Thereby, the algorithm achieved promising results comparing to the dermatologists involved in the study, proving to be capable of assist physicians in the classification of lesions with a melanocytic origin.

Han *et al.* [202] used a Microsoft ResNet-152 architecture in order to develop an algorithm able to classify 12 different types of skin lesions. Besides, the authors implemented a Grad-CAM method to better understand the prediction of the CNN considering gradient-based localization [203]. Three datasets were taken into account for the evaluation of the proposed model's performance (Asan, Edinburgh and Hallym), comprising a total of around 19 thousand images. The AUC values corresponding to the first one were 0.96, 0.83, 0.82, and 0.96 for the diagnosis of BCC, squamous cell carcinoma, intraepithelial carcinoma, and melanoma, respectively, whereas the achieved AUC values for the same classes in the Edinburgh test dataset were of 0.90, 0.91, 0.83, and 0.88, respectively.

In the approach proposed by Marchetti *et al.* [204] for lesion classification, all of the automated predictions resulting from the participation of 25 teams in the ISBI 2016 Challenge were fused into a unique classification system, using 5 different fusing methods. These methods comprised both non-learned approaches, as prediction score averaging and voting, and machine learning procedures, namely greedy ensemble fusion, linear binary SVM, and non-linear binary SVM. The algorithms were tested using 100 dermoscopic images of different lesions. In order to evalute the algorithm's performance, similarly to the work proposed in [15], the results were also compared to the classification provided by dermatologists. Greedy fusion was the best-performing fusion algorithm, achieving a ROC area of 0.86 against the mean ROC area of 0.71 corresponding to the 8 dermatologist's performance.

Regarding lesion classification, Bi *et al.* [205] followed three different approaches. In the first one, the original three-class problem was accessed, while the second one comprised two binary classification problems that allowed to distinguish melanoma vs. others and seborrheic keratosis vs. others, respectively. Concerning the third approach, it consisted of an ensemble of the two other approaches in order to obtain the classification results. In all of them, pre-trained ResNet architectures were fine-tuned, having the number of neurons in the last layer been modified to match the number of classes. As such, the three CNNs were evaluated, resulting in an average AUC of 90.60%, 91.30%, and 91.50% for the three approaches, respectively.

Besides the previously proposed work of Kawahara *et al.* [200], the authors also developed a method using end-to-end learning for lesion classification [206]. The developed CNN architecture was composed of several tracts that allowed to learn simultaneously information of an image at different resolutions, employing hybrid pre-trained AlexNet networks. The CNN comprised an end layer that incorporates the outputs corresponding to the different image resolutions to a single layer. Also, auxiliary supervised loss layers were added to each tract, in order to regularize the results. The algorithms' performance was tested on Dermofit Image Library, being the best average accuracy achieved of 79.5%.

Sun *et al.* [207] introduced a new dataset for real-world skin lesions classification which comprises 198 different categories of lesions. Apart from hand-engineered features, the authors also

applied CNNs for skin lesion classification. Regarding the classification based on deep features, a pre-trained CNN model was considered, having the weighting parameters be fine-tuned. Moreover, in the test phase, a comparison between the CaffeNet model for deep feature extraction in combination with an SVM classifier and the pre-trained VGGNet model for image classification was made. This last one provided a best average accuracy value over all 198 classes, corresponding to 50.27%. However, when compared to the performance of hand-crafted features extraction, which achieved an accuracy of 52.19%, these results showed to be worst.

Also Lopez *et al.* [208] proposed a method based on the VGGNet convolutional neural network architecture. The work comprised three different approaches for skin lesion classification, whose performance was compared. The first approach consisted of training the CNN from scratch; in the second one, the transfer learning paradigm was employed, freezing convolutional blocks and only training the fully-connected layers of a pre-trained VGGNet; finally, in the third approach, transfer learning was also considered, having been made a fine-tuning of the weighting parameters of the pre-trained VGGNet network. The three presented methods were tested on the ISBI 2016 Challenge dataset and the best testing result corresponded to an accuracy of 81.33%, associated with the last approach.

Nasr-Esfahani *et al.* [209] developed a CNN from scratch in order to distinguish between melanoma and benign cases. To achieve this goal, the clinical images used as input to the neural network were firstly pre-processed and data augmentation was also performed, due to the limited amount of available images. The proposed CNN comprises two convolutional layers followed by pooling layers and a linear transfer function is used to predict the diagnosis. Regarding the performance of the proposed algorithm, an accuracy of 81% was obtained.

3.3 Object Detection in Dermatology

In the field of dermatology, many works comprising object detection have been developed, namely with the purpose of localizing skin lesions in images. In this section, two of the proposed methods are presented.

Vesal *et al.* [210] developed a framework for skin lesion detection that is then used to assist segmentation. Firstly, a network similar to Faster-RCNN was used to localize the lesion in the skin and generate a bounding box around the proposed region. This network comprises shared convolution layers for feature extraction, a region proposal network that defines the anchor boxes and provides the probability of a lesion is contained in the proposed box, and also an R-CNN that improves the predicted regions of interest, providing the bounding box coordinates and classifying if the lesion is present or not. Then, the generated regions of interest are used as input for SkinNet [211], a modified version of U-Net also proposed by Vesal *et al.*, in order to allow skin lesion segmentation. The achieved results were promising, having outperformed other state-of-the-art methods for lesion segmentation, with a Jaccard index of 88%.

Also Qian *et al.* [212] proposed a methodology for first detecting and then segmenting skin lesions in dermoscopy images, under the scope of ISIC Challenge 2018. For the detection process,

Mask R-CNN was applied in order to obtain the RoI of the lesion. The images were cropped not exactly by the bounding box, but with a random expansion and contraction. Then, they were used as input for a network that contains an atrous spatial pyramid pooling (ASPP) block to extract information in different scales and to provide the segmentation map of the lesion. This method was the winner of the challenge, obtaining a Jaccard index of 80.2%.

Although a few studies have been presented, several methods concerning object detection were already developed in dermatology [213][214]. However, these works are mainly focused on skin lesion detection, not aiming to localize other objects, such as rulers, which are sometimes considered as artifacts [215][216].

3.4 Incremental Learning

The problem of catastrophic forgetting in machine learning algorithms has led to the development of different methodologies that aim to enable models to acquire new knowledge incrementally, maintaining information related to previous tasks. As previously mentioned, these strategies comprise three main categories, namely: architectural strategies, regularization strategies, and rehearsal strategies. Since these categories do not contradict each other, it is possible that they are used in combination, resulting in hybrid strategies. Thus, some of the developed works concerning the three mentioned strategies will be addressed in this section.

3.4.1 Architectural strategies

Learn++ [168] was proposed by Polikar *et al.* and is inspired by the adaptive boosting (AdaBoost) algorithm. This model consists of an ensemble of weak classifiers that are trained with different training sets, according to a distribution. These classifiers are considered as weak since the prediction accuracy is close to random guessing. However, through a boosting procedure, these weak learners are transformed into effective classifiers, being the final classification obtained by combining the outputs of the different learners through weighted majority voting. Due to the implementation of weak classifiers, the algorithm converges faster and overfitting issues are eliminated.

Another architectural strategy was developed by Roy *et al.* [22]. The proposed system consists of a tree-structured CNN architecture, as presented in Figure 3.1, that hierarchically grows when new classes are introduced. The structure is extended by adding new leaves dependent on the similarities between the old and the new classes. Therefore, the knowledge is transmitted through the nodes of the structure. Although this approach allows the attenuation of catastrophic forgetting problem, as the Tree-CNN continues growing with time, more memory and computational costs are required.

Rusu *et al.* [217] proposed Progressive Neural Networks (PNNs). This approach allows to transfer knowledge across sequential tasks, without forgetting previously learned information. PNNs maintain a pool of pre-trained models that, through lateral connections, enable the extraction of knowledge that will be considered in the following tasks, as presented in Figure 3.2a. For

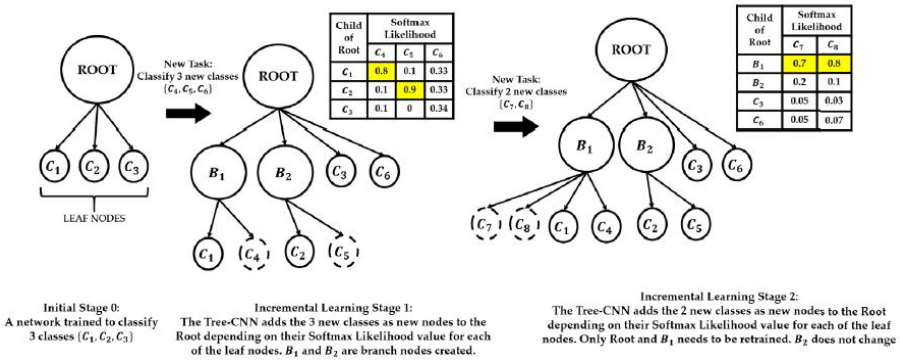
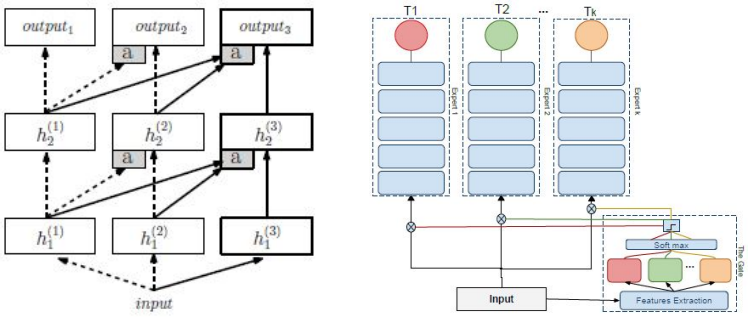


Figure 3.1: Incremental learning stages of the Tree-CNN [22].

each task a new neural network is created which is connected to all of the preceding tasks, preserving the knowledge acquired since the beginning. However, similarly to the previously presented model [22], this can also be a limitation, as the number of parameters to be considered by each task will gradually increase.

Expert Gate [218], proposed by Aljundi *et al.*, allows to learn and add new tasks to the model based on what was already learned. However, in this approach, data from previous tasks (experts) are not stored, which consists of an advantage when comparing to the models presented before. Instead, auto-encoder gates are considered, which are able to produce outputs similar to the inputs. Therefore, the encoder allows the selection of the most relevant expert for the new task by learning its representation and, at the test phase, directly forwards it to the proper expert. A schematic diagram of this strategy may be founf in Figure 3.2b.



(a) Progressive Neural Networks representation [217].

(b) Expert Gate representation [218].

Figure 3.2: Representation of some architectural strategies for incremental learning.

3.4.2 Regularization strategies

Concerning regularization strategies, Kirkpatrick *et al.* developed the Elastic Weight Consolidation (EWC) [219]. Depending on the importance of the weights in the preceding tasks, the system slows down the learning process. In this way, it is possible to selectively constrain the update of parameters, ensuring that no relevant variations are verified in important weights when the model

is fine-tuned on new tasks. The importance of the weights is evaluated through the Fisher information matrix, which is also considered in the loss function of the new task.

Synaptic intelligence (SI) was presented by Zenke *et al.* [171], consisting of an approach similar to EWC. Nevertheless, instead of computing the weights importance offline by means of the Fisher information, in SI, the relevance of weights (synapses) is evaluated online, during stochastic gradient descent processing. Therefore, by identifying the relevant synapses, the algorithm prevents them from suffering alterations in future tasks. This way, are the least important synapses that intervene in the learning of new tasks, avoiding catastrophic forgetting. The process implies a modified loss function where it is also considered a quadratic surrogate loss and a dimensionless strength parameter. Despite the implementation differences, when comparing SI performance to EWC, both approaches yield similar results.

Learning without Forgetting (LwF) is another regularization strategy to tackle catastrophic forgetting proposed by Li *et al.* [220], being one of the first methods to consider knowledge distillation for incremental learning. This approach takes into account the outputs of the previous model to learn new tasks, which works as a regularizer for the new learning. Basically, before the new task is trained, its data is trained on the old classifier. The generated predictions are then considered during the train phase in order to limit the update of the parameters on the new task via distillation. Therefore, the loss function comprises not only the popular cross-entropy loss function but also a knowledge distillation loss component, which intends to preserve a stable response.

3.4.3 Rehearsal and pseudo-rehearsal strategies

Rebuffi *et al.* developed iCarl (Incremental Classifier and Representation Learning) [221], a model intended for class-incremental learning. For each already observed class, iCarl retains a set of samples. Being so, since the current training set comprises both stored and new examples, it allows to transfer knowledge from the previous to the new tasks. The classification is based on the nearest-mean-of-exemplars, which consists of the selection of the class that presents the nearest distance to prototypes. The update of the parameters is based on the minimization of a modified loss function which also comprises a distillation loss. For this reason, iCarl may be considered as an hybrid strategy.

FearNet was proposed by Kember *et al.* [222], inspired by the mammalian hippocampal complex. FearNet is composed of three networks, as presented in Figure 3.3: one for long-term storage (mPFC), another one for recent memories (HC), and the other one that defines which of the previous networks is required for prediction (BLA). Differently from iCarl, in FearNet, previous examples are not stored, increasing the memory efficiency of the system. Nevertheless, pseudorehearsal strategy is used, allowing to take into account previous memories without storing them and enabling the consolidation of recent memories. Catastrophic forgetting is mitigated by the use of an autoencoder. During the sleep phase, the autoencoder generates pseudo-examples using a Gaussian distribution, allowing to train mPFC to accommodate the inputs stored in HC. Therefore,

the generation of previously learned instances in combination with the new samples are used to fine-tune mPFC for memory consolidation.

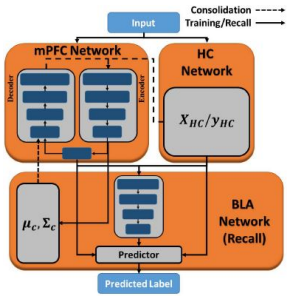


Figure 3.3: FearNet modules [222].

Another rehearsal strategy is Gradient of Episodic Memory (GEM) [177] that was developed by Lopez *et al.*. In this system, a subset of already seen patterns for each class is stored. The goal of the model is to minimize the negative backward transfer, this is, the decrease on the model's performance when a new task is considered. Moreover, contrary to what is verified in other strategies that prevent important weights to be update, in GEM the authors allow positive backward transfer, which tolerates the weights to change in case it leads to an increased performance on some of the preceding tasks. A-GEM [223] is also a rehearsal strategy and consists of an upgraded and more efficient version of GEM. In this strategy, it is intended to ensure that the average loss of the episodic memories over the previous tasks does not increase. To achieve this goal, the dot product between a reference gradient corresponding to the average of the previous tasks and the gradient of the current task is considered. In case it is negative, the gradient is projected on the current task. Comparing to other strategies such as EWC [219] or iCarl [221], the A-GEM strategy presented the best trade-off between efficiency and accuracy.

Chaudhry *et al.* [224] explored a rehearsal approach, dubbed Experience Replay, that combines both examples of the current tasks and some random examples that are stored in an external memory in the training batches. The authors also evaluated different methods to populate the memory: reservoir sample, ring buffer, k-Means, and mean of features. In the case of the reservoir sample, having a memory buffer with size "*memsize*" and being "n" the number of already seen points, each data point is sampled with a probability of $\frac{memsize}{n}$. Concerning the ring buffer approach, it assumes a FIFO (first-in-first-out) buffer for each class, which contains $\frac{memsize}{num\_classes}$ samples. This way, it ensures equal representation of all classes in the memory, storing the last samples of each class. With the k-Means approach, k centroids are estimated for each class. Then, the examples whose feature representation is closer to those selected centroids are the ones stored in the memory. Regarding the mean of features, the average feature vector is computed before the classification layer and the examples whose feature representation is closer to this vector are stored. The results showed that Experience Replay is able to outperform other strategies such as EWC or A-GEM. Concerning the methodology to populate the memory, the reservoir sample obtained the best performance for bigger episodic memories, while ring buffer performed better for tiny episodic memories.

3.4.4 Incremental learning in Medical context

In the last years, some approaches implementing incremental learning techniques for medical applications have been proposed, namely for classification and segmentation tasks.

Meng *et al.* [225] proposed ADINet, an incremental system for retinal image classification. This approach considers both class label prediction and attribute prediction in order to improve classification performance. The authors assume that the attribute prediction of each class works as an encoder for models representation. Hence, knowledge distillation is used to retain knowledge from the previous classes, as well as an attribute distillation loss that allows to constrain the attribute prediction for the previous and new models. The classification of the image label is made simultaneously with the attribute prediction, which involves the calculation of the attributes weights in each image, taking into account the information entropy of each attribute. Thereby, the overall loss function combines the distillation loss and the classification loss which takes into account both the loss function for image-level classification and the loss function for attribute classification. The system achieved an accuracy of 82.7% using a fundus image dataset, outperforming some state-of-the-art approaches, as is the case of iCarl [221] or LwF [220].

Ravishankar *et al.* [226] developed an incremental learning method based on feature transformers whose performance was then demonstrated on two medical applications: X-ray pneumothorax classification and ultrasound cardiac view classification. In this approach, besides the extracted features being mapped into a new representation by adding dense layers, the classification loss is modified by the addition of a center-loss, which ensures class-wise separation. In a first approach, the experiments comprised both multi-task and single incremental task settings of the iCIFAR100 dataset, and were then demonstrated in real-life medical problems. In the multi-task setting, the proposed method outperformed all of the considered state-of-the-art methods, such as EWC [219], iCarl [221], or GEM [177], whereas in the single incremental task iCarl performed better. Concerning the pneumothorax classification, a pretrained VGG network was implemented, considering the ChestXRay dataset. The features to take into account comprised the ones extracted from two pooling layers and fully connected layers. After each step, a dense layer was added in order to map the features into a new representation, consisting of the feature transformers. The validation accuracy after each layer was evaluated, resulting in a lower value for the deepest layers, due to the increased fine-tuning. In terms of the cardiac view classification, four common views of adults and pediatrics images were considered. In the first task, only two adult views were used, being then considered the pediatrics images of the same view in order to infer the domain adaptation of the system. In the following task, the other two views were added to simulate new task learning. It was verified that alterations in the domain do not affect the model's performance and that it can learn new tasks without the problem of catastrophic forgetting.

Since in clinical practice the protocols and policies used to acquire images are continuously changing, Hofmanninger *et al.* [172] implemented a rehearsal method that considers a dynamic memory in order to mitigate forgetting when CT data is obtained using different scanner protocols. The authors started by finetuning a pretrained ResNet50 that is continuously being updated with

new data. To retain information from one task to another, some previously seen data (dynamic memory) is stored and together with the new examples is used to create the new training data. The memory update is made according to some established criteria. This update depends on a high level metric based on the gram matrix, in order to ensure that previous cases are not replaced if they are visually distant. The proposed approach was compared to other strategies, such as the EWC [219], showing to be more efficient in dealing with forgetting. Different memory sizes were also explored. On the one hand, it was verified that increasing the memory size can reduce the catastrophic forgetting. On the other hand, this increase slows the adaptation to new tasks.

For glioma segmentation from MR imaging, Garderen *et al.* [227] employed the previously presented EWC system [219], in order to evaluate its performance on the desired purpose. The segmentation task involved a 3D U-Net neural network in different scenarios: one network to train both domains (source and target) at the same time for 100 epochs, and another network that trained firstly the source domain (composed by low and high-grade glioma) for 50 epochs, and then the target domain (containing non-enhancing low-grade glioma) also for 50 epochs. This was performed with and without EWC on the loss function. Dice score results showed that the use of EWC allowed an improvement of performance in the source domain, but a decrease in the target domain. Therefore, it is concluded that EWC, on the one hand, allows to prevent catastrophic forgetting, but, on the other hand, limits the ability of the model to adapt to a new domain.

Similarly to Garderen *et al.* [227], also Baweja *et al.* implemented EWC [219] in order to perform incremental segmentation in brain MRI. The segmentation comprised two tasks: task A was related to normal structures, namely cerebrospinal fluid, grey matter, white matter, while task B consisted of white matter lesions. In this work, the DeepMedic 3D network was considered, and, once again, the results showed that EWC allows to mitigate catastrophic forgetting, preserving the performance of the first task.

Also Karani *et al.* [228] developed an incremental method for brain MR segmentation. This approach consisted of using a CNN that shares convolutional filters and batch normalization layers which consider specific parameters for each domain. Different datasets were considered and divided in five domains, from which two domains were not used from the beginning. The authors implemented a U-Net network with slight alterations to achieve the intended goal, since batch normalization layers can be included in any CNN. The results showed that the proposed method lead to an increase of the Dice score for the old domains.

That being said, it is notorious that some effort has been done in the last years in order to avoid catastrophic forgetting and allow models to be trained incrementally. However, there is still a long way to go, since these strategies present some limitations and only a few studies were performed in medical context. Concerning regularization approaches, due to the constant addition of loss terms, it is possible that after several tasks there is a saturation of the model, which can compromise the tasks performance. On the other hand, in the case of rehearsal approaches, these may require a separate memory which can also be considered as a limitation. Therefore, it is seen that incremental learning is a challenging part of artificial intelligence systems, and for that reason, new systems that ensure robustness, and flexibility of the models continue to be required.

Chapter 4

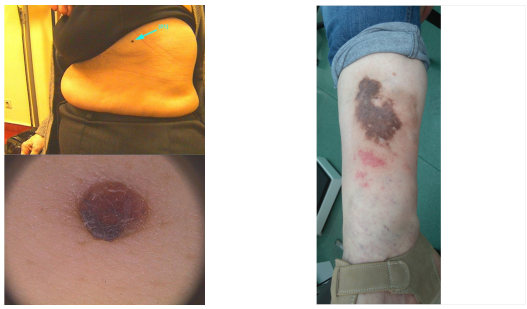
Methodology

As previously introduced, this dissertation aims to achieve two major goals. On the one hand, classification and object-detection algorithms should be developed and implemented in order to evaluate what is the most suitable approach to predict if an image contains a ruler or not. On the other hand, various models able to accurately classify dermatological images according to their modality should be developed. These models must explore different incremental learning techniques to allow them be updated as new images are available, without losing performance on the already trained images and without the need of having access to all of previous images to retrain the model from scratch. Therefore, this chapter comprises three main sections. Section 4.1 covers the dataset that was employed during the development of this work; Section 4.2 presents both the classification and object-detection algorithms that were exploited; and, finally, the models and incremental learning strategies that were implemented with the aim of classifying images according to their dermatological modality are presented in the last section (Section 4.3).

4.1 Dataset

To achieve the intended goals, a dataset containing a total of 5203 dermatological images from the Portuguese National Health System from retrospective data related to the referral requests from Local Health Care Units for the first Dermatology Hospital consultation in the scope of DermAI project was firstly considered. This dataset is composed by several anatomic, dermoscopic, macroscopic, and full-body images, some of them containing a ruler in order to allow the inference of the lesion size in teledermatological consults.

The first steps of the work comprised the preparation of these images since some of them were duplicated, contained two photos (in some cases from different modalities), or included white frames that could interfere with the results, as shown in Figure 4.1. Hence, the repeated images were removed, the photos were separated, and the white padding was eliminated, respectively. Moreover, namely in the dermoscopic class, images where the ruler was imperceptible due to blur issues were removed too. After this preparation steps, the dataset was composed by a total of 4955 images.



(a) (b)

Figure 4.1: (a) Example of image containing two photos corresponding to different modalities: the upper photo belongs to the anatomic modality and the bottom one to the dermoscopic modality; (b) Example of image with white padding.

Besides the aforementioned modalities of images, primary-care clinicians may also send other clinical information for teledermatological consults, such as medical reports. Due to confidentiality issues, this data could not be directly used in the work and, for this reason, images that could represent this information had to be acquired from other sources. To accomplish this purpose, a *Chrome* extension called "Imageye" [1] was used in order to collect this class of images from *Google Images*. The search included terms as "medical report", "clinical report", "report", "medical form" and others, resulting in a total of 1020 images. In summary, the amount of images corresponding to each dataset's class is presented in Table 4.1.

Table 4.1: Dataset composition according to images' modality and presence of ruler.

	No Ruler	**Ruler**	**Total**
Anatomic	1502	75	1577
Dermoscopic	1052	245	1297
Full-body	352	1	353
Macroscopic	1209	519	1728
Clinical reports	1020	0	1020
Total	5135	840	5975

Although the dermatological images contained in the dataset are associated with an anonimysed patient ID, in some cases, the same patient has more than one ID by mistake. Therefore, in order to avoid biased results when splitting the dataset, besides grouping images that corresponded to the same patient ID, a **K-means** algorithm was implemented with the aim of clustering similar images. The algorithm employs a pre-trained VGG-16 network for feature extraction, whereby the last layer of the CNN was not considered in order to only obtain a feature vector from each image. After applying PCA to reduce the features vectors' dimensionality, these were used as the K-means input. Different numbers of clusters were considered, according to the number of images

[1] `https://chrome.google.com/webstore/detail/image-downloader-imageye/agionbommeaifngbhincahgmoflcikhm`

associated with each dermatological modality, to ensure that all similar images were assigned to the same set.

The new dataset was split into three different sets for training, validation, and testing purposes in a proportion of 60:20:20, respectively. Since one of the goals of the work involves incremental learning techniques, 900 images were stored in order to later feed the algorithms intended for image modality classification, maintaining the proportion of each class. Hence, the dataset distribution after the splitting process can be found in Table 4.2. Besides, in Figure 4.2, some examples concerning the different modalities may be found.

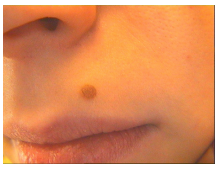
Table 4.2: Dataset after splitting process.

	Train		**Validation**		**Test**		**Incremental**		
	No Ruler	**Ruler**	**No Ruler**	**Ruler**	**No Ruler**	**Ruler**	**No Ruler**	**Ruler**	**Total**
Anatomic	766	37	255	13	255	13	226	12	1577
Dermoscopic	536	125	179	41	179	42	158	37	1297
Full-body	179	1	60	0	60	0	53	0	353
Macroscopic	616	264	206	88	205	89	182	78	1728
Clinical report	521	0	173	0	173	0	153	0	1020
Total	2618	427	873	142	872	144	772	127	5975

(a) Anatomic.

(b) Dermoscopic.

(c) Full-body.



(d) Macroscopic.

(e) Clinical report.

Figure 4.2: Examples of images from the different modalities.

4.2 Ruler inference

As previously mentioned, some of the images that are considered in teledermatological consults contain a ruler that allows dermatologists to infer the size of a skin lesion. In order to predict the presence of a ruler in different modalities of dermatological images, several classification and object-detection algorithms were implemented (Sections 4.2.1 and 4.2.2, respectively) and the performance of these two types of methods' was then compared.

Here, the goal was only to know if an image contained a ruler or not and, for this reason, nor the images intended for incremental learning, nor the clinical reports were taken into account. Moreover, images' modalities were not distinguished, and only two classes were considered: images with a ruler and images without a ruler, as presented in Table 4.3. The data distribution was preserved, maintaining a proportion of 60:20:20 for training, validation, and testing.

Table 4.3: Dataset considered for ruler inference.

	No Ruler	Ruler	Total
Train	2097	427	2524
Validation	700	142	842
Test	699	144	843
Total	3496	713	4209

4.2.1 Classification algorithms

As introduced in Chapter 3, the efficiency of traditional machine learning algorithms is highly dependent on the extracted features that are used to train them, which may not be the most suitable ones. Moreover, the selection of the most proper features is a complex and time consuming process. For this reason, the preference for deep learning approaches in the last years have been remarkably increasing. These models have already proved their effectiveness in the image classification field, being able to perform feature extraction by themselves and to achieve promising performance results.

Hence, for this binary problem, three different approaches using CNNs were considered to accomplish the feature extraction and subsequent image classification. The first approach consisted of a neural network that was conceived from scratch, whereas the other two approaches employ transfer learning using a VGG-16 architecture pre-trained on the ImageNet database [126]. The implementation of these models was made using the Tensorflow API 2.4.1 in Python 3.7.1 on a CPU with 16GB of RAM. Concerning the developed CNN, it is composed by three convolutional and max-pooling layers, including batch normalization (BN) and dropout techniques too, in order to stabilize the learning process and avoid overfitting, as presented in Figure 4.3. In the case of the VGG models, one of the approaches consisted of fine-tuning the already pre-trained model by unfreezing and continuing to train all of the layers, which allows to better learn the patterns specific to the considered dataset. In the other approach, feature extraction was performed using the pre-trained model and only the last layers were adapted to this classification problem, leveraging generic features that are shared among different images and optimizing the process for the intended purpose. Hereinafter, the first VGG approach is called "FT VGG-16", whereas the second one is simply called "VGG-16".

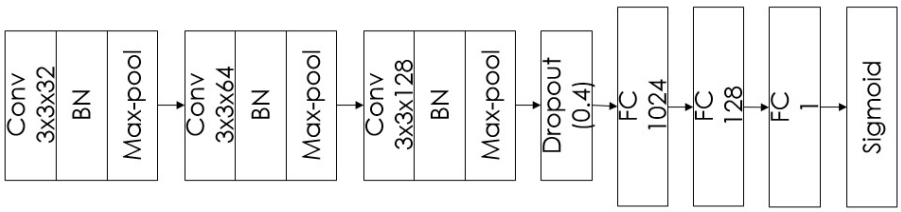


Figure 4.3: Representation of the developed CNN.

Due to the high class imbalance that was verified between the two classes (Table 4.3), an oversampling of the training images corresponding to the class with less amount of images, this

is, the set of images containing a ruler, was performed. This oversampling was made offline, involving horizontal and vertical flipping with a probability of 0.5, and also random channel shifts in the range [0,55], and alterations in brightness in the percent range of [0.4,0.8], resulting in a total of 1996 images with ruler during the training phase. The process did not include rotations nor cropping, since as in some images the rulers were localized in the periphery, these techniques could have removed them.

As deep learning algorithms can handle raw data, the pre-processing of the images only comprised their normalization, and resizing since the dataset was composed by a wide variety of images' sizes. Thus, pixel intensities were transformed to fit between the range 0 and 1, and images were resized to 224x224 pixels. It is worth noting that, after the oversampling and resizing, all images were verified in order to ensure that no damaging alterations that could compromise the visibility of the rulers had been applied.

In the three models, different approaches were exploited in order to optimize the algorithms and assess the influence of different parameters in the learning process. After slight variations on the hyperparameters, the three models were trained for 20 epochs using a batch size of 16, the sigmoid classifier, and considering the Adam algorithm to optimize the process. Considering that the learning range defines how much the weights are updated at each step, this fundamental hyperparameter was optimized. Hence, different values were tested in the range of $1x10^{-3}$ to $1x10^{-7}$ and, in the case of the CNN trained from scratch, it was set to $1x10^{-7}$ and to $1x10^{-6}$ in the transfer learning models. These were the chosen values because when higher learning rates were considered, some volatility on the validation curves were verified, especially in the case of the simple CNN. This could indicate that the employed learning rates were resulting in a high modification in the weights, providing very different predictions on the validation examples. For this reason, the corresponding values were decreased to the mentioned ones.

Since the aim of these algorithms was to predict if a ruler was presented in the images or not, the amount of false detections is an aspect that must be taken into account. Thus, beyond the standard binary cross-entropy loss function (Equation 4.1), a weighted cross-entropy loss function that aims to penalize both false positives and false negatives was also applied. This function is given by Equation 4.2, which introduces weights both in the cost of a false positive and of a false negative.

$$J_{bce} = -\frac{1}{M}\sum_{i=1}^{M}[y_i * log(h_\theta(x_i) + (1-y_i) * log(1-h_\theta(x_i))] \tag{4.1}$$

$$J_{wbce} = -\frac{1}{M}\sum_{i=1}^{M}[w_{fn} * y_i * log(h_\theta(x_i) + w_{fp} * (1-y_i) * log(1-h_\theta(x_i))] \tag{4.2}$$

In these equations, M represents the number of training examples, y_i is the ground truth for the training example i, h_θ represents the model with weights θ, x_i is the input for the training sample i, and w_{fn} and w_{fp} represent the cost of a false negative over a true positive and the cost of a false positive over a true negative, respectively. As the aim was to compare the influence of the

loss function in the amount of false detections, different sets of weights were applied as it will be further discussed in the next chapter.

Since the performance of deep learning algorithms depends on the amount of data available to train them (Figure 4.4), the influence of data augmentation was also explored, using techniques similar to the ones used in the oversampling task. This augmentation was made online during the training process, where at each epoch, every image was randomly transformed.

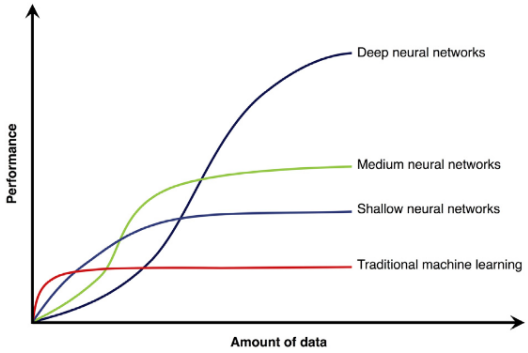
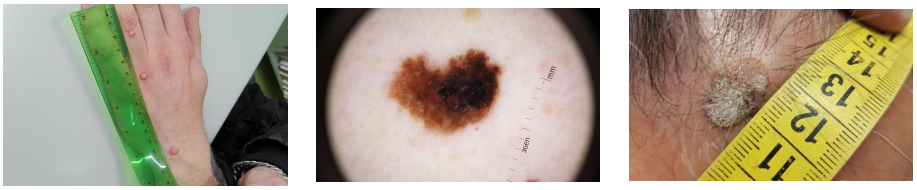


Figure 4.4: Impact of the amount of available data on the performance of traditional machine learning and deep learning algorithms [229].

Finally, as presented in Figure 4.5, the rulers contained in dermoscopic images differ a lot from traditional rulers that are used in the other modalities. Therefore, the considered models were trained with and without this modality of images, allowing to understand if this class could interfere with the results.



(a) Anatomic image. (b) Dermoscopic image. (c) Macroscopic image.

Figure 4.5: Differences in rulers depending on image's modality.

An overview of the most relevant approaches that were applied to the models in order to optimize their learning process is presented in Table 4.4.

Table 4.4: Approaches considered for classification model's selection.

	Approach
Loss function	Binary Cross-Entropy Weighted Cross-Entropy
Data augmentation	With augmentation Without augmentation
Dataset	With dermoscopic class Without dermoscopic class

4.2.2 Object-detection algorithms

For training and testing the object-detection models, the Tensorflow Object Detection API [2] was used. This repository contains the implementations of several state-of-the-art algorithms and it is possible to find models pre-trained on the COCO dataset [230]. Within the scope of this work, Tensorflow 2.4.1 in Python 3.7.10 on a NVIDIA T4 GPU with 8GB of memory was considered.

As the dataset used in the work did not contain annotations regarding the rulers' localization, firstly, all images that contain a ruler had to be labelled. The annotations were made using an intern platform of Fraunhofer AICOS that allowed to obtain the coordinates of the rulers' bounding-boxes by computing the coordinates of the upper left corner and its width and weight.

In the object detection API, data is read using a TFRecord file format, which must comprise the bounding-boxes' normalized coordinates (x/width, y/height) defined by four floating-point numbers [ymin, xmin, ymax, xmax], as well as the class of the objects, and the encoded RGB images themselves. Thus, it was necessary to create three files in order to train, validate, and test the models.

Three state-of-the-art algorithms, already described in Chapter 2, were explored to detect the existence of rulers in the images. More specifically, this selection comprised the EfficientDet-D0 [148], the RetinaNet [137], and the Faster R-CNN [144] detectors. The EfficientDet-D0 was selected because it is part of a new family of detectors, reporting high efficiency and accuracy; the RetinaNet and the Faster R-CNN were also considered because they are very popular detectors, able to achieve promising results in the literature. All of them were pre-trained on the COCO dataset and trained for 3000 steps. Although some minor changes have been made, namely concerning the batch size and the data augmentation techniques, all of the other training variables were similar to the original implementation.

Concerning the EfficientDet-D0, this detector uses the EfficientNet-B0 as the backbone network, requiring an input size of 512x512 pixels. Differently from the original implementation where besides horizontal flipping also image cropping is performed for data augmentation, in this work only the first technique was employed. Similarly to what was previously mentioned concerning the classification algorithms (Section 4.2.1), this decision had to do with the fact that in some cases the rulers are located in the periphery of the images, whereby the crop could have removed them. Also, due to computational costs, a batch size of 16 had to be used in this model, since this was the maximum size that did not raise memory errors.

In the case of the RetinaNet, the authors reported that using a ResNet-101 architecture as the backbone network led to better results than when using a ResNet-50 [137]. Therefore, in order to allow a comparison of the detector's performance depending on the employed backbone, these two networks were considered, as presented in Table 4.5. Similarly to the previously mentioned detector, data augmentation during training only employed horizontal flipping techniques too for the same reasons, and the batch size was set to 8, as, in this case, it was the maximum size for this model with the available memory.

[2] https://github.com/tensorflow/models/tree/master/research/object_detection

Regarding the Faster R-CNN, in the original paper it is also mentioned that using the ResNet-101 architecture as the network for region proposal instead of the VGG-16 increases the system performance. However, due to the associated computational costs, only the ResNet-50 network was employed in this detector. In the same way as the aforementioned models, data augmentation only comprised horizontal flipping techniques and the batch size was also set to 8 because of the same reasons.

In Table 4.5 it is possible to find a summary referring to the previously mentioned alterations concerning each one of the detectors.

Since in some images the input sizes required by the detectors differ a lot from their original sizes, in order to ensure that the rulers were not affected by this resizing and were still visible for such dimensions, before feeding the detectors, all images were resized (for 640x640 and 512x512 pixels), stored, and individually analysed.

Table 4.5: Object detection model's configuration.

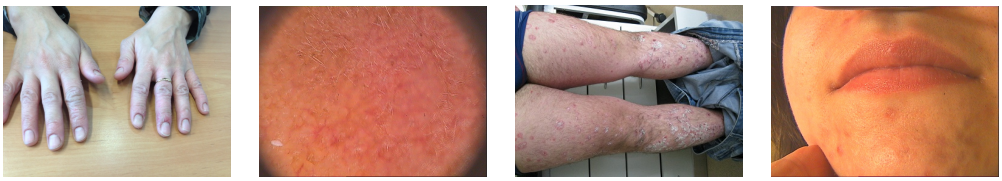
	Backbone net.	**Input size**	**Batch size**	**Data aug.**
EfficientDet-D0	EfficientNet-B0	512	16	Horiz. flipping
RetinaNet	ResNet-50	640	8	Horiz. flipping
	ResNet-101	640	8	Horiz. flipping
Faster R-CNN	ResNet-50	640	8	Horiz. flipping

4.3 Image Modality Classification

4.3.1 Data preparation

Since one of the aims of incremental learning is to allow algorithms to adapt to new conditions, the dataset considered in the previous phase was readjusted in order to simulate the presence of the concept-drift, already described in Section 2.7.1. To achieve this, some particular types of images were only used on the incremental phase, as shown in Figure 4.6. In the case of the anatomic modality, all images that contained hands or feet were only considered in the second phase of the learning process. Concerning the dermoscopic modality, the images selected for the second task comprised the ones that presented a pink coloring, like the one presented in Figure 4.6b. Regarding the full-body modality, this selection refers to images where legs and arms are presented, and in the case of macroscopic images, images that contained regions of the face were only used in the incremental phase.

The distribution of the dataset considered in this part of the work can be consulted in Table 4.6, where a proportion of 60:20:20 was maintained for training, validation and testing, respectively, in the first task. With regard to the 900 images intended for the incremental task, these were split in a proportion of 80:20 for training and testing, respectively. To simplify the comprehension in some situations throughout this document, the first task is also called task A, and the incremental part corresponds to task B.



(a) Anatomic. (b) Dermoscopic. (c) Full-body. (d) Macroscopic.

Figure 4.6: Types of images only considered in the incremental phase for each modality.

Table 4.6: Dataset distribution for image modality classification before oversampling.

	First task (A)			Incremental task (B)	
	Train	**Validation**	**Test**	**Train**	**Test**
Anatomic	803	268	268	190	48
Dermoscopic	661	220	221	156	39
Full-body	180	60	60	43	10
Macroscopic	880	294	294	209	51
Clinical reports	521	173	173	123	30
Total	**3045**	**1015**	**1016**	**721**	**178**

In this table, it is possible to verify the imbalance that exists across the five modalities. For this reason, an oversampling of the training images corresponding to the less representative classes (i.e. dermoscopic, full-body and clinical reports) was also applied in this phase, resulting in around 4000 images related to the first task and around 1000 in the case of the incremental task. Once again, this oversampling was made offline, including techniques, such as horizontal and vertical flipping with a probability of 0.5, alterations in brightness with a percent value $p \in [0.4,0.8]$, zoom shifts using a percent value $p \in [0.8,1.2]$, and width shifts in the percent range of [-0.15,0.15]. After employing these techniques, all images were assessed to ensure that they have not gone through damaging alterations. Similarly to what was previously mentioned (namely, in Section 4.2.1), after this procedure, the images' pre-processing only comprised normalization and resizing approaches, due to the nature of deep learning algorithms.

4.3.2 Base models selection

After preparing the data, different models that could accurately classify images' modalities were developed in order to be later used as the base models for the incremental learning process. The implementation of these models and of the following work was made adopting the PyTorch API 1.8.1 in Python 3.7.10 on a NVIDIA T4 GPU with 8GB of memory. The choice relied on two different networks, allowing to understand the differences verified in the behaviour of the considered incremental learning strategies according to the complexity of the models. Therefore, the VGG-16 architecture was chosen because it is a very popular and effective network used for image classification purposes and, as reported in the previous chapter (Section 3.1.2), it has already

been employed for image modality classification in other medical fields; and the MobileNetV2 architecture since it is a smaller network able to achieve great results in image classification too.

Both networks were used in a transfer learning approach, having been previously trained on the ImageNet dataset. The pre-trained networks were used as feature extractors, being a new set of layers added to the top of the extracted features which were trained to this modality classification problem. Thus, in the case of the VGG-16 network, around 12 million trainable parameters were considered, whereas in the MobileNetV2 model, only about 650 thousand parameters were trainable.

The models were optimized and the selected configurations are presented in Table 4.7. In both of them, the Adam optimizer was considered and a Cross-Entropy loss function was applied.

Table 4.7: Configuration of the selected base models.

Model	Epochs	Batch size	LR	Dropout
VGG-16	50	16	$1x10^{-5}$	Yes (0.5)
MobileNetV2	50	16	$1x10^{-5}$	Yes (0.4)

Considering these configurations, each model was trained three times using merely the images corresponding to the first task. The models' parameters after each of the three runs were stored to allow a later comparison of the results, and only the run that provided the best outcomes for each network was later used as a starting point for the incremental task, corresponding to the training of task A.

4.3.3 Incremental Learning of Image Modalities

To implement different incremental learning approaches, the Alpha version of the *Avalanche* library, which is an end-to-end open-source library, specifically designed for incremental learning, was adopted [231]. This decision had to do with the fact that, usually, incremental learning algorithms have to be implemented from scratch, using different assumptions and settings, which makes it difficult to compare their performance, even when the same benchmarks are considered. Besides, due to the fast-growing interest in incremental learning, often different terminologies are considered. Hence, the *Avalanche* allows to favor the flexibility and simplicity of incremental learning implementations, without employing a strict nomenclature.

4.3.3.1 Incremental Learning approaches

Besides the Naïve and the Cumulative approaches which were used as baseline strategies (Section 2.7.3.1), three incremental learning approaches were explored. The choice relied on the EWC [219], AGEM [223], and Experience Replay [224] since these are popular strategies that were previously considered in medical context, as presented in Section 3.4.4.

As previously presented (Section 3.4.2), being the EWC a regularization strategy, it introduces a new term in the loss function that aims to penalize alterations in the most important weights of

the following tasks. The importance of the weights is given by the diagonal of the Fisher matrix, which measures the amount of information carried by a given variable, in this case, the trainable variables of the model. Thus, the function that is intended to be minimized in the incremental training is given by:

$$L(\theta) = L_B(\theta) + \sum_i \frac{\lambda}{2} F_i(\theta_i - \theta^*_{A,i})^2 \tag{4.3}$$

where $L_B(\theta)$ is the loss corresponding to the incremental task only, λ is an hyperparameter that sets how important the previous task is compared with the new one, F represents the diagonal of the Fisher matrix, θ_i is the set of weights and bias of the current (incremental) task, $\theta^*_{A,i}$ represents the sets of weights and bias of the previous task, and i labels each parameter.

Concerning the AGEM and the Experience Replay, both of them are rehearsal strategies, which means that some information from the first task is maintained in the memory and will then be trained together with the new task.

In the case of AGEM, this strategy considers a fixed memory to store patterns from a previous task. A reference gradient is then computed, consisting of the average of the gradients from a random set of examples contained in this memory. If the dot product between this reference gradient and the gradient of the current task is negative, the gradient is projected via Equation 4.4, ensuring that the loss over the previous tasks does not increase.

$$\tilde{g} = g - \frac{g^T g_{ref}}{g^T_{ref} g_{ref}} g_{ref} \tag{4.4}$$

In this equation, g refers to the gradient of the current task and g_{ref} is the reference gradient.

In Experience Replay, on the other hand, a random subset of images from the previous task which are contained in an external memory is concatenated with the incremental dataset at each training batch. The examples considered in each batch are balanced, ensuring an equal number of images from the various tasks.

These strategies require that some variables are tuned, in order to be implemented. Regarding EWC, the λ hyperparameter must be set to ponder the penalization that should be assigned to the loss function. The higher this value, the greater the regularization that will be applied. In what concerns the rehearsal strategies (AGEM and Replay), the size of the memory buffer must be adjusted.

Therefore, the chosen values were based on values reported in the literature (EWC: [223],[173]; AGEM: [223]; Replay: [232]) and can be found in Table 4.8.

A wide range of λ values is usually employed in the case of the EWC approach because it is an hyperparameter that must be set according to the problem. Concerning the AGEM strategy, although other memory sizes are also encountered in the literature, the maximum memory size that was possible to explore within the scope of this work was 150, due to memory limitations. With respect to the Replay strategy, on the other hand, since the incremental task only comprised around 1000 training images (after oversampling), it was decided to restrict the memory size to

500 examples from the previous tasks, although larger memory sizes are also reported on the literature.

Table 4.8: Parameters used in the incremental learning approaches.

Strategy	Parameter
EWC	$\lambda = 100$
	$\lambda = 50$
	$\lambda = 1$
	$\lambda = 0.5$
AGEM	Mem = 50
	Mem = 100
	Mem = 150
Replay	Mem = 100
	Mem = 250
	Mem = 500

4.3.3.2 Incremental task training

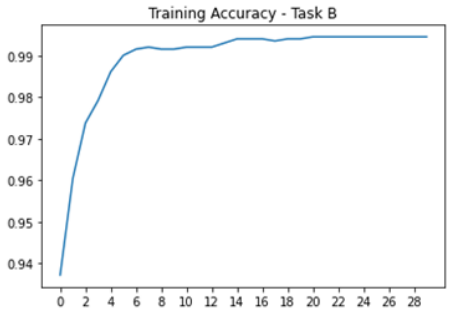
In order to establish a more reliable comparison between the performance of the different incremental learning strategies, the code of the *Avalanche* library had to be slightly adjusted to prevent the training of the first task and to employ the models previously trained on this task. Thus, the training of task B started from the same point for the various incremental strategies.

Similarly to the base models training (Section 4.3.2), the previously mentioned learning rates were considered, the Adam optimizer was used and the Cross-Entropy loss function was employed in this training. Concerning the batch size, this had to be set to 8 due to the involved computational cost. Moreover, in the case of the number of epochs, a lower number was also considered. As presented in Figure 4.7, it is verified that although the incremental training affects the learning process of the MobileNetV2 in terms of stability, in the case of the VGG-16 model, it converges after around 20 epochs. For this reason and also because training models sequentially is usually prone to catastrophic forgetting, it was evaluated if training longer had an influence on the performance of the first task. To do so, the second task was trained for 10, 20, and 30 epochs, allowing to understand the alterations namely in what concerns the forgetting.

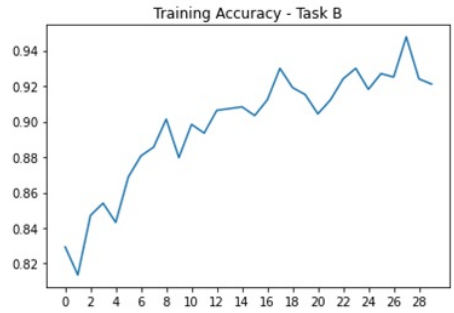
For both models, each configuration was implemented for 10 iterations, in order to provide more robust and reliable results.

4.3.3.3 Incremental Learning evaluation

As it was possible to understand by the literature review that was made, there is still no consensus among the computer vision community in what concerns the evaluation of the incremental learning strategies' performance. However, it was verified that this assessment typically relies on the accuracy's computation at different levels (among the different tasks or global performance) and on the efficiency of the models. Therefore, within the scope of this work, the evaluation of the



(a) Example of the VGG-16 model considering the Naïve approach.



(b) Example of the MobileNetV2 model considering the Naïve approach.

Figure 4.7: Accuracy evolution during the training of the incremental task (Task B).

employed incremental learning strategies was based on these two main aspects: the accuracy and the efficiency.

In what concerns the accuracy, the BWT and the FWT metrics, previously introduced on Section 2.8, were implemented. These metrics require the computation of an accuracy matrix (R), which is given in Table 4.9. Basically, the performance of the model after it finishes learning about the training task Tr_i is evaluated on all test tasks Te_i, even the ones referring to future tasks. Hence, for instance, $R_{B,A}$ refers to the test accuracy of the task A after the model have been trained on task B.

Table 4.9: Accuracy matrix R. Tr_i = training; Te_i = testing.

R	Te_A	Te_B
Tr_A	$R_{A,A}$	$R_{A,B}$
Tr_B	$R_{B,A}$	$R_{B,B}$

The higher the BWT and the FWT, the better. Besides, a negative BWT is usually related to catastrophic forgetting, which means that the performance of the previous task decreased after performing the incremental training.

Furthermore, also the accuracy of all testing images (task A and task B together) after the two training processes was computed. This evaluation was made in order to evaluate the performance of the obtained models on a set of examples containing images belonging to the two employed distributions.

Concerning the efficiency of the different approaches, for every incremental learning strategy, both the time required to train each epoch and the maximum RAM memory used throughout the learning process were assessed. The evaluation of the RAM memory usage was made every 0.5 seconds and the maximum value that was reached was considered.

Chapter 5

Results and Discussion

This chapter presents the obtained results and the corresponding analysis concerning the two major goals of this dissertation: the ruler inference (Section 5.1) and the incremental image modality classification (Section 5.2).

5.1 Ruler inference

To predict whether a ruler is contained in different dermatological images or not, several classification and object-detection algorithms were developed and implemented in order to ascertain the most advantageous configuration in solving this task. Therefore, in this section, the achieved results for each type of models are presented.

5.1.1 Classification algorithms

As mentioned in the previous chapter (Section 4.2.1), three different approaches were considered for this binary classification problem: a simple CNN conceived from scratch, a fine-tuned VGG-16 network previously trained on the ImageNet database (FT VGG-16), and a VGG-16 network pre-trained on the same database that was used for feature extraction. These three approaches were chosen allowing to compare the performance of a simpler network with a more complex one that have already proven its effectiveness in image classification problems, as seen in Chapter 3, and also to compare two transfer learning methodologies.

In order to establish the most suitable models for the intended goal, different strategies were exploited. The influence of the employed loss function on the number of false detections was studied considering two different functions. Firstly, a binary cross-entropy was used in the three models and then a new weighted cross-entropy loss function was implemented. In this last case, different weights were assigned to the false negatives and to the false positives to evaluate their impact on the learning process. The achieved results for the most relevant sets of weights can be consulted in Table 5.1. Nevertheless, other weight values were also tested.

By looking over these results and in what concerns the fine-tuned VGG, it is possible to infer that, although the number of false detections had decreased when the weighted cross-entropy

Table 5.1: Ruler classification algorithms results depending on the loss function.

Model	Loss Function	Weights	FN	FP	TN	TP	Acc.
Simple CNN	Binary Cross-Entropy	—	27	87	612	117	0.865
	Weighted Cross-Entropy	**FN: 1; FP: 2**	40	30	669	104	**0.917**
	Weighted Cross-Entropy	FN: 1; FP: 4	60	13	686	84	0.913
	Weighted Cross-Entropy	FN: 1.5; FP: 4	50	36	663	94	0.898
FT VGG-16	Binary Cross-Entropy	—	4	5	694	140	0.989
	Weighted Cross-Entropy	**FN: 1; FP: 2**	3	3	696	141	**0.993**
	Weighted Cross-Entropy	FN: 1; FP: 4	3	5	694	141	0.991
	Weighted Cross-Entropy	FN: 1.5; FP: 4	4	3	696	140	0.992
VGG-16	Binary Cross-Entropy	—	11	15	684	133	0.969
	Weighted Cross-Entropy	**FN: 1; FP: 2**	11	12	687	133	**0.973**
	Weighted Cross-Entropy	FN: 1; FP: 4	31	2	697	113	0.961
	Weighted Cross-Entropy	FN: 1.5; FP: 4	14	10	689	130	0.972

function was considered, the difference on the results depending on the employed sets of weights was not very significant and with the binary cross-entropy it was already possible to achieve a good performance.

In the case of the other pre-trained VGG and of the simple CNN, on the other hand, the difference in the results depending on the employed loss function and among the different sets of weights was more notorious. When the weighted cross-entropy loss function was used and only the false positives were penalized with a weight of 4, although this function allowed to decrease the number of false positives, it is also possible to notice an increase in the number of false negatives, resulting in fewer true positives. Nevertheless, when the weight associated with a false negative was set to a value different from 1 (it was increased to 1.5) and the penalization of a false positive was kept in 4, it may be verified that despite the number of false positives had increased compared to the previous setting, the growth in the number of false negatives is less pronounced compared to the results achieved using the binary cross-entropy function, allowing to obtain more true positives. Moreover, when the penalization of a false positive was twice as the weight associated with a false negative, the results seem to be more balanced in terms of false detections (number of false positives and false negatives). Comparing the three sets of weights, this is the setting that provides less amount of false detections in the three models. With respect to the simple CNN, for instance, although the number of false negatives had increased compared to the binary cross-entropy, this increase was not significant when looking at the total number of false predictions which dropped from 114 (in the case of the binary cross-entropy) to 70 when the weighted cross-entropy was taken into consideration.

Therefore, it is possible to infer that, when compared to the binary cross-entropy, the weighted cross-entropy loss function allowed to decrease the number of false detections (false negatives plus false positives) in the three models and for almost all of the three weights configurations. Also, it is possible to see that when the weighted cross-entropy loss function is used, there is the need of establishing a trade-off between the number of false positives and false negatives, which may be tuned taking into account the ratio between the associated weights.

That being said, through the analysis of these results, the weighted cross-entropy loss function was selected for the three models, considering a weight of 1 and of 2 to penalize the false negatives and the false positives, respectively.

As previously mentioned, the influence of data augmentation on the results was also explored, since the amount of data used to train the models may influence the outcomes. Hence, the same model configuration was used to evaluate the alterations verified in the models' performance. In Table 5.2, it is possible to find the obtained results concerning the three models.

Table 5.2: Ruler classification algorithms results with and without data augmentation, using weighted cross-entropy with 1:2 weights.

Model	Approach	Acc.	Prec.	Sens.	Spec.	AUC
Simple CNN	Without aug.	**0.917**	**0.776**	**0.722**	**0.957**	**0.952**
	With aug.	0.875	0.617	0.715	0.908	0.923
FT VGG-16	Without aug.	0.988	0.965	0.965	0.993	0.998
	With aug.	**0.993**	**0.979**	**0.979**	**0.996**	**0.998**
VGG-16	Without aug.	**0.973**	0.917	**0.924**	**0.983**	**0.994**
	With aug.	0.968	**0.921**	0.889	0.984	0.993

By analysing the results contained in this table, it may be inferred that in the context of this problem, data augmentation does not provide any benefit to the VGG model used for feature extraction, nor to the simple CNN. In fact, in the majority of the metrics, the performance of these two models decreased when this approach was considered. This may be due to an increased difficulty of the problem during the training phase resulting from the different patterns that had to be learned. Nevertheless, in the case of the fine-tuned VGG, there is an improvement on the algorithm's performance when more data is considered during the training process. A possible explanation for this fact relies on the higher amount of parameters that had to be tuned in the learning phase, since, in this case, the whole pre-trained model was unfreezed to better adjust to the proposed problem. For this reason, more data was required in order to allow the model to learn effectively.

Another study intended to evaluate how the dermoscopic modality could compromise the results due to the differences verified in the rulers' appearance. Therefore, the same models were trained with and without the images corresponding to this class. Taking into account the results obtained in the previous study, data augmentation was applied to the FT VGG-16. As it is possible to see in Table 5.3, the results for all models did not improved when the dermoscopic class was not considered. This may result from the smaller amount of images considered during the training phase, since the dermoscopic modality represents more than 20% of all images contained in the dataset, as presented in Table 4.2. Thus, it is possible to infer that this modality does not affect the performance of the models, and consequently, it may be maintained in the dataset without penalizing the results.

Moreover, both the results presented in Table 5.2 and in Table 5.3 show that the VGG models outperformed the simple CNN model in all of the considered metrics for the two scenarios (with and without data augmentation, and with and without the dermoscopic class). This can be

Table 5.3: Ruler classification algorithms results with and without the dermoscopic modality.

Model	Dataset	Acc.	Prec.	Sens.	Spec.	AUC
Simple CNN	With dermo	**0.917**	**0.776**	**0.722**	**0.957**	**0.952**
	No dermo	0.878	0.649	0.556	0.941	0.918
FT VGG-16	With dermo	**0.993**	**0.979**	**0.979**	**0.996**	**0.998**
	No dermo	0.980	0.934	0.944	0.987	0.994
VGG-16	With dermo	**0.973**	**0.917**	**0.924**	**0.983**	**0.994**
	No dermo	0.955	0.874	0.844	0.976	0.983

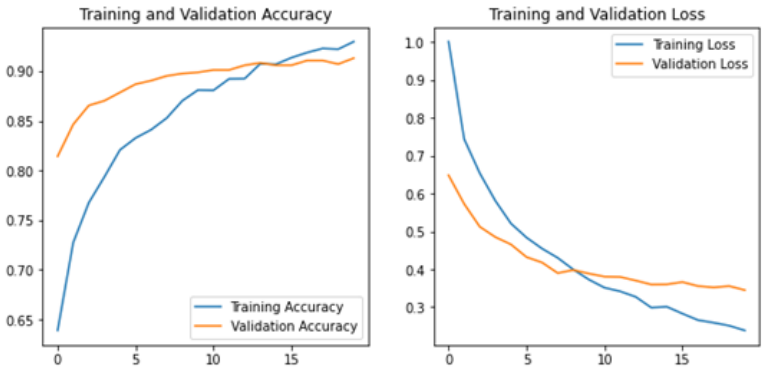
explained by the fact that these models employ transfer learning and, for this reason, even with limited data, they were able to achieve good results, since they had already been trained on a larger dataset, which enabled them to learn generic features common to every images. Another possible explanation relies on the simplicity of the CNN trained from scratch, since its low complexity may have prevented the model to learn representative features. Furthermore, it is also possible to compare the performance of the two VGG approaches that take into account transfer learning. The fine-tuned VGG surpassed the pre-trained VGG network for feature extraction. This result was more or less expected since with fine-tuning, on the one hand, the model could already identify generic features, such as edges or textures and, on the other hand, was further able to learn features more intrinsic to the dermatological dataset that were not previously seen in the ImageNet database and that the pre-trained VGG for feature extraction was not able to adjust to.

Therefore, considering the comparison on the results when using different approaches, the configurations of the selected classification models are presented in Table 5.4.

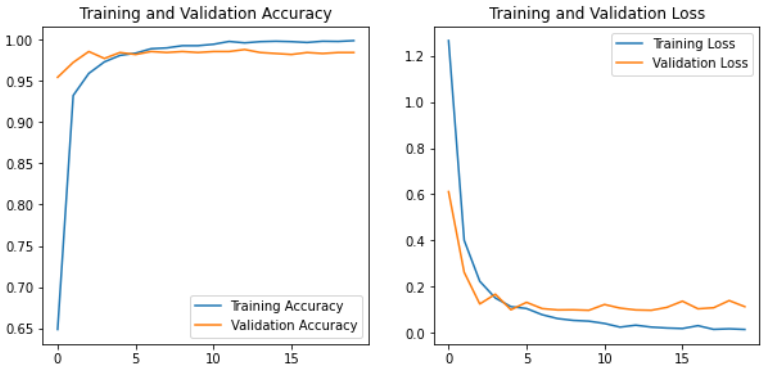
Table 5.4: Selected classification models.

Model	Loss function	Weights	Dataset	Data aug.
Simple CNN	Weighted	FN: 1; FP: 2	With dermo	No
FT VGG-16	Weighted	FN: 1; FP: 2	With dermo	Yes
VGG-16	Weighted	FN: 1; FP: 2	With dermo	No

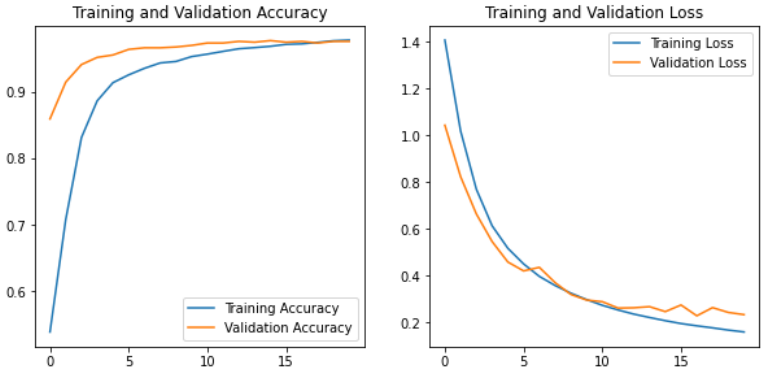
To ensure that these models were reaching the desired behavior, the curves corresponding to the learning process were plotted and can be found in Figure 5.1. In the case of the VGG model pre-trained for feature extraction (Figure 5.1c), it can be seen that during the entire process, the validation accuracy surpassed the one achieved during the training phase. This may result from the fact that regularization techniques, such as dropout, are only employed during the training phase but not in the validation phase. This phenomenon may also result from the oversampling that was done in order to balance the number of training images belonging to each class, which could have increased the complexity of the problem. Nevertheless, it is possible to verify that for the three models the loss function decreased throughout the learning process, whereas the accuracy increased, both in the case of the validation and the training phases, as supposed.



(a) Simple CNN.



(b) Fine-tuned VGG.

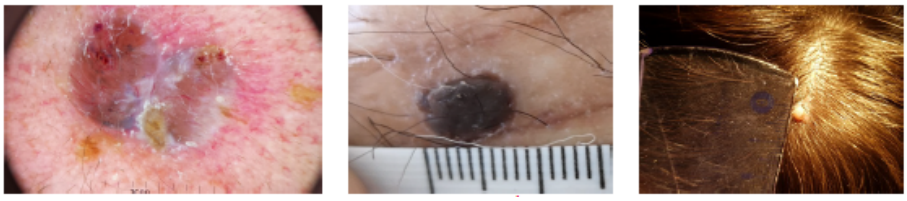


(c) VGG as feature extractor.

Figure 5.1: Learning curves of the three selected models.

The predictions provided by the classification algorithms were also visually inspected. In Figure 5.2, the misclassifications of the fine-tuned VGG can be consulted. The upper row comprises the false positives whereas the bottom row contains the false negatives. With respect to the identified false negatives, as presented in Figure 5.2b, its difficulty in recognizing rulers localized in the periphery of the images was demonstrated. Regarding the pre-trained VGG as feature extractor, some examples of the wrong predictions can be found in Figure 5.3. In what concerns the images related to the false positives, it is possible to verify that they essentially comprise images where straight edges may be found.

(a) False positives.



(b) False negatives.

Figure 5.2: Misclassifications of the fine-tuned VGG.

(a) False positives.

(b) False negatives.

Figure 5.3: Misclassifications of the pre-trained VGG as feature extractor.

5.1.2 Object-detection algorithms

Concerning the object-detection algorithms, different detectors were employed allowing to compare the performance of both one-stage and two-stage detectors. The selection relied on the EfficientDet-D0, RetinaNet, and also Faster R-CNN, as introduced in Section 4.2.2 of the previous chapter.

A common metric that is used to evaluate the performance of the detectors in terms of the spatial location of the predictions is the mean Average Precision (mAP). This metric is based on the Intersection over Union (IoU) (Table 2.5) which allows to compute the overlap between the

predicted and the ground-truth bounding boxes. Establishing a threshold, it is then possible to infer if the provided detection is a true detection or not, enabling to estimate the number of true positives (TP), false positives (FP), and also false negatives (FN). In other words, if $IoU > threshold$, the prediction is considered as a TP; on the other hand, if $IoU < threshold$, it is a false positive; if no prediction is made but a ground-truth bounding box exists, it is classified as a false negative. Based on these values, the Precision and Recall metrics (Table 2.5) may be computed, allowing to obtain the precision-recall curve of each model. Therefore, the Average Precision (AP) corresponds to the area under this curve. In some cases the AP is computed for each class, and is then averaged to get the mAP. However, in other cases, as in the COCO evaluation challenge, there is no difference between these metrics [230]. Moreover, as in this work it is just intended to find rulers in the images, only one class is considered, and for this reason, AP and mAP correspond to the same metric.

The mAPs were computed for two different IoU thresholds: 0.5 and 0.75. In Table 5.5, it is possible find the achieved test results for both of them.

Table 5.5: Object-detection algorithms comparison.

	Backbone net.	mAP_{50}	mAP_{75}
EfficientDet-D0	EfficientNet-B0	0.940	0.760
RetinaNet	ResNet-50	0.144	0.022
	ResNet-101	0.361	0.085
Faster R-CNN	ResNet-50	**0.977**	**0.913**

By analysing the results, Faster R-CNN achieved the best results compared to the other detectors, both when using a threshold of 0.5 and of 0.75. Being a two-stage detector, it was expected to achieve high accuracy and to be slower than the one-stage detectors, what was verified. However, the results do not totally match with what was reported in the literature, since usually, RetinaNet is able to achieve better results than both Faster R-CNN and EfficientDet-D0 [137][148]. In this case, this was not verified with the ResNet-50 backbone, nor when using the ResNet-101, having its performance fell short of expectations. This may be due to the employed hyperparameters that, as will be further analysed in Section 5.1.3, may have not been completely optimized.

Since the aim of this phase of the work was to compare the performance of the object-detection algorithms with the classification ones in predicting if an image contains a ruler or not, besides the mean Average Precision that allows to compare the detectors among them, other metrics were also considered, as presented in Table 5.6. As each image contains at most one ruler and the models output multiple detections for the same object, only the predicted bounding-box with higher confidence was maintained. In this case, the amount of TP, TN, FP, and FN in the test phase was computed considering an IoU threshold of 0.5. The metrics were then calculated using the equations presented in Table 2.5.

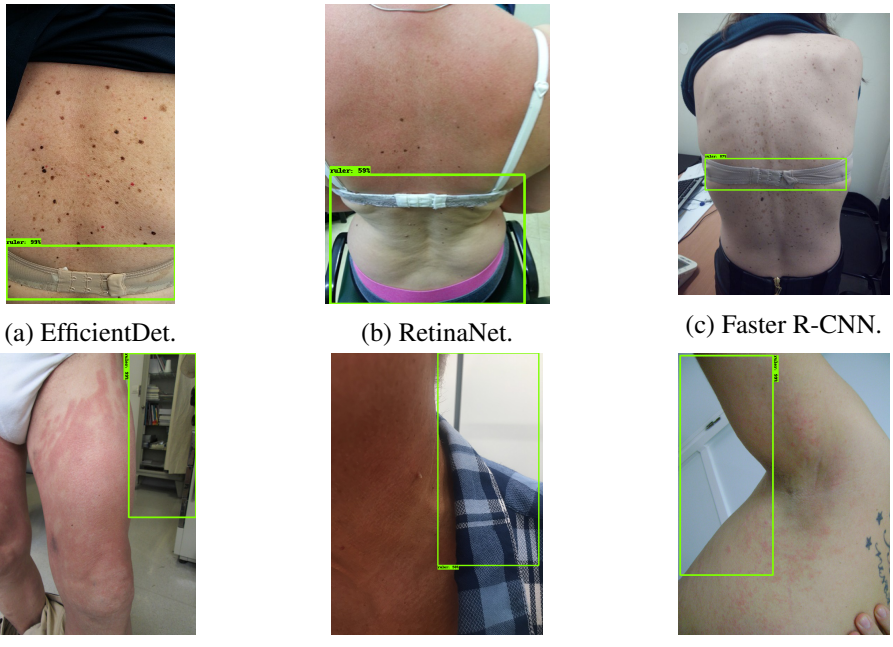
By looking at Table 5.6, once again, it is possible to see that Faster R-CNN outperformed the other detectors in almost all of the considered metrics. Furthermore, on the one hand, the difference between the RetinaNet results when considering the ResNet-50 or the ResNet-101 as

Table 5.6: Object-detection results.

	Backbone net.	Acc.	Prec.	Sens.	Spec.
EfficientDet-D0	EfficientNet-B0	0.836	0.506	0.908	0.822
RetinaNet	ResNet-50	0.848	0.667	0.210	**0.979**
	ResNet-101	0.848	0.647	0.231	**0.974**
Faster R-CNN	ResNet-50	**0.925**	**0.698**	**0.979**	0.914

the backbone network is not marked, and beyond that, the associated computational cost when using this last network was higher, taking more time to perform the training process. Hence, in the context of this work, it was not advantageous to use the ResNet-101 network. In the next section (Section 5.1.3), these results will be further analysed and compared with the ones achieved using the classification algorithms.

The predictions of the three detectors were then visually analysed. In these images, the bounding-boxes plotted in blue correspond to the ground-truth bounding-boxes, whereas the green boxes are the predicted ones. Some common errors were identified among the three detectors. For instance, both in the case of the EfficientDet and of the Faster R-CNN, the detectors considered that some of the bras' back strips were a ruler, as shown in Figure 5.4 on the upper line. Also, it was verified that some objects contained in the background were being identified as rulers, possibly due to the presence of straight edges, as presented in the same figure on the bottom line.



(a) EfficientDet. (b) RetinaNet. (c) Faster R-CNN.

(d) EfficientDet. (e) RetinaNet. (f) Faster R-CNN.

Figure 5.4: Visual results of the object-detection algorithms - False Positives.

Moreover, specially concerning the rulers contained in the dermoscopic images, when these were localized in the periphery of the image, the EfficientDet algorithm could not detect them.

Some examples of this issue are presented in Figure 5.5. The RetinaNet, on the other hand, was not able to detect any ruler contained in the dermoscopic images.

Figure 5.5: Visual results of the object-detection algorithms - False Negatives of EfficientDet.

Concerning the Faster R-CNN, only three rulers were not identified by this detector, being these images shown in Figure 5.6. The image on the far right was misclassified by the three detectors perhaps due to the shape and color of the ruler which is not as evident as in the other images. This image was also not predicted by the fine-tuned VGG, as previously presented in Figure 5.2.

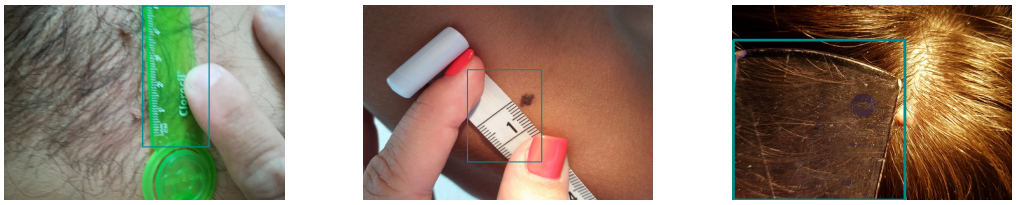
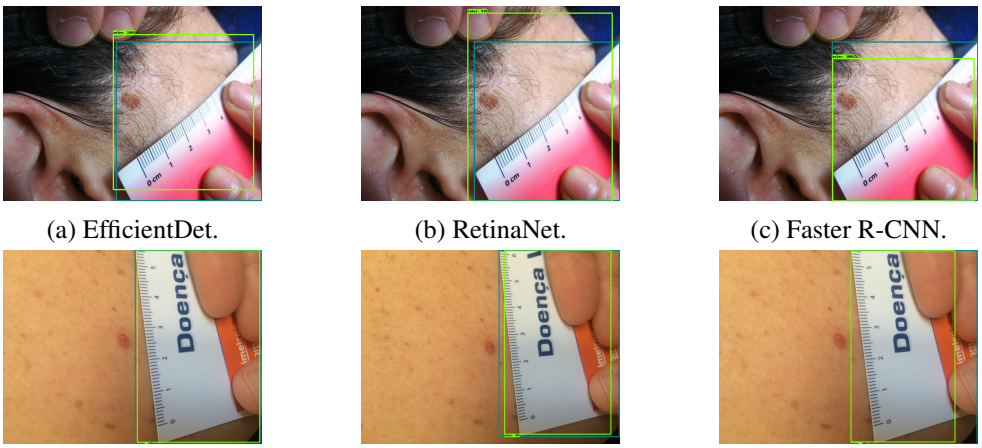


Figure 5.6: Visual results of the object-detection algorithms - False Negatives of Faster R-CNN.

Besides, in Figure 5.7 some examples of rulers that were correctly localized by the three detectors can be found. Other examples of the algorithms' predictions may be consulted in the Appendix on Section B.1.



(a) EfficientDet. (b) RetinaNet. (c) Faster R-CNN.

(d) EfficientDet. (e) RetinaNet. (f) Faster R-CNN.

Figure 5.7: Visual results of the object-detection algorithms - True Positives.

5.1.3 Algorithms comparison

In Table 5.7, the best results concerning the classification and object-detection algorithms are presented.

Table 5.7: Comparison of the best classification and object-detection algorithms.

	Acc.	**Prec.**	**Sens.**	**Spec.**
FT VGG-16	**0.992**	**0.979**	0.972	**0.996**
Faster R-CNN	0.925	0.698	**0.979**	0.914

Considering what was previously discussed, it is possible to infer that, in general, the classification algorithms were able to achieve better results comparing to the object detection ones. Some possible explanations were found that may be responsible for the lower object-detection models' performance. For instance, the fact that these models require a rectangular bounding-box, and, in some situations, depending on the ruler's position, it may occupy most of the image, leading to the introduction of noise in the training phase and to an increased difficulty of the problem. An example of this issue is represented in Figure 5.8, where the corresponding bounding-box is represented in blue and includes practically the whole image.

Figure 5.8: Example of a bounding-box (blue) that occupies almost the entire image.

Besides, the object detection algorithms were pre-trained on the COCO dataset [230], which contains more than 200 thousand images belonging to 80 different object categories. However, although these categories comprise common objects, none of them contains rulers. Furthermore, in this dataset the number of small objects is higher than the number of large objects. Around 41% of the objects as an area smaller than 32x32 pixels and only 24% corresponds to objects larger than 96x96 pixels in terms of the associated segmentation mask [230]. Taking this into account, since the dimensions of the rulers contained in dermatological images are mostly higher than 96x96 pixels, the fact that the models have been pre-trained on this dataset may have compromised the results.

Then, it is also important to mention that since the main goal was not to precisely localize rulers in the image, but only to infer if they were present or not, the algorithms were not completely optimized, having been used almost all of the original settings of the detectors. Especially in the case of the RetinaNet detector, the employed configuration may not have been the most suitable one.

Moreover, it is noteworthy that all of the considered object-detection models were trained using only one GPU provided by Fraunhofer. However, the original implementations of RetinaNet and Faster R-CNN report the usage of 8 GPUs for the training process [137][144]. Thus, the achieved object-detection results may indicate that further training was required to obtain results more similar to the ones reported in the literature.

As a conclusion, considering all of the explored approaches and the obtained results, the employed classification models proved to be effective in predicting if an image contain a ruler or not. Even a simple CNN is able to tackle this binary problem successfully, whereby it is possible to infer that this is a relatively simple classification task. Therefore, in order to predict if an image contain a ruler or not, the classification algorithms are preferable over the object-detection models, specially when transfer learning is considered.

5.2 Image Modality Classification

5.2.1 Base models selection

As introduced in Section 4.3.2 of the previous chapter, after preparing data, two different architectures were explored in order to define the base models corresponding to the training of the first task. Therefore, only the images belonging to task A were considered in these implementations. The models were trained three times using the same configurations and only the train that provided the best results for each architecture was considered. The results corresponding to the selected models can be found in Table 5.8, whereas the results concerning the other two runs of the VGG and of the MobileNetV2 models can be found in the Appendix C.1.

Table 5.8: Results of the selected base models tested on task A.

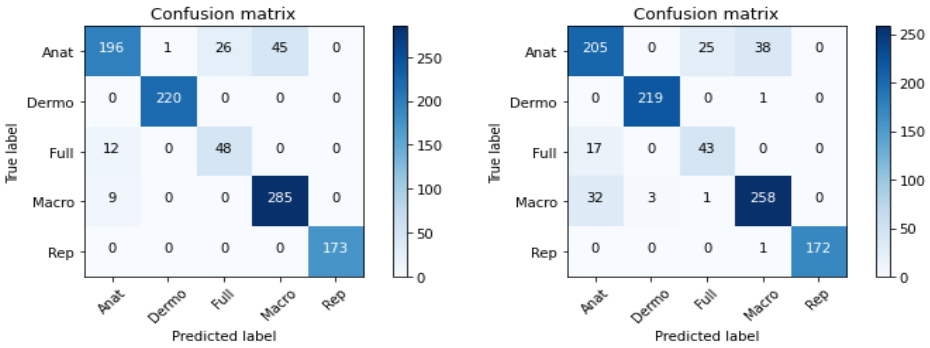
Model	Modality	Accuracy	Precision	Recall	F1-score
VGG-16	Anatomic	**0.9084**	0.9032	0.7313	0.8082
	Dermoscopic		0.9955	1.0000	0.9977
	Full-body		0.6486	0.8000	0.7164
	Macroscopic		0.8636	0.9694	0.9135
	Clinical reports		1.0000	1.0000	1.0000
	Macro average	—	0.8822	0.9001	0.8872
	Weighted average	—	0.9132	0.9084	0.9070
MobileNetV2	Anatomic	0.8837	0.8071	0.7649	0.7854
	Dermoscopic		0.9865	0.9955	0.9910
	Full-body		0.6232	0.7167	0.6667
	Macroscopic		0.8658	0.8776	0.8716
	Clinical reports		1.0000	0.9942	0.9971
	Macro average	—	0.8565	0.8698	0.8624
	Weighted average	—	0.8850	0.8837	0.8840

As it is possible to verify by the results presented in this table, both in the case of the VGG-16 model and of the MobileNetV2 model, the modality that most negatively influenced the results was

the full-body one, taking into account the F1-score. This may be due to the smaller variability of images belonging to this category since the full-body modality was the less representative class of the original dataset where only 180 examples were available for training (Table 4.6). Nevertheless, although an oversampling has been made in order to balance the classes during the training phase, this technique is not as efficient as considering different images, due to the fact that a small feature diversity is introduced.

In addition, comparing the results achieved by the two models, the VGG-16 model surpassed the performance of the MobileNetV2 in what concerns the four computed metrics. These outcomes may result from its higher complexity, being able to better identify features intrinsic to each modality.

In order to reinforce these results and provide their visual analysis, the confusion matrices corresponding to the two models were plotted and can be found in Figure 5.9.



(a) VGG model. (b) MobileNetV2 model.

Figure 5.9: Confusion matrices of the two selected models.

Through these matrices, it may be verified that some anatomic, full-body, and macroscopic images were confused by the two models; some examples of these misclassifications are presented in Figure 5.10. It is worth noting that in some cases, the images belonging to these classes are very similar, being difficult to effectively differentiate them. Hence, as the labeling of the images was manually made by various people, it is possible that different labels have been assigned to identical images, as shown in Figure 5.11, which may have influenced the results. However, in general, both the VGG-16 and the MobileNetV2 were able to correctly predict the modalities of the different dermatological images, which is represented by the darker shades on the diagonal of the matrices.

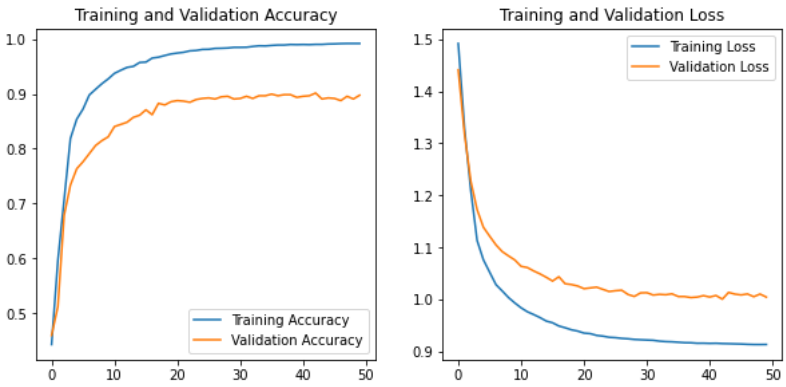
(a) Anatomic image classified as full-body. (b) Anatomic image classified as macro. (c) Full-body image classified as anatomic. (d) Macroscopic image classified as anatomic.

Figure 5.10: Images misclassified by the two selected models.

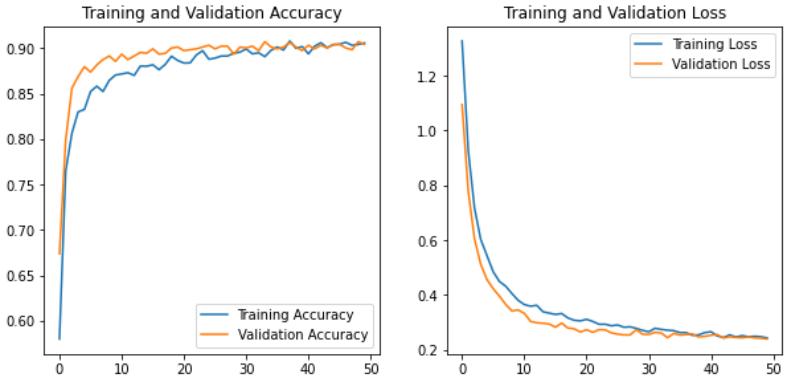
(a) Anatomic (left) | Full-body (right). (b) Anatomic (left) | Macroscopic (right).

Figure 5.11: Examples of similar images belonging to different modalities.

Moreover, to ensure that the models were achieving the desired behaviour, also the learning curves corresponding to the accuracy and the loss throughout the training and validation phases were plotted and are represented in Figure 5.12. In both cases, it is possible to verify the convergence of the models besides an increase on the accuracy and decrease on the loss as more epochs are concluded.



(a) VGG-16 model.



(b) MobileNetV2 model.

Figure 5.12: Learning curves of the two selected models.

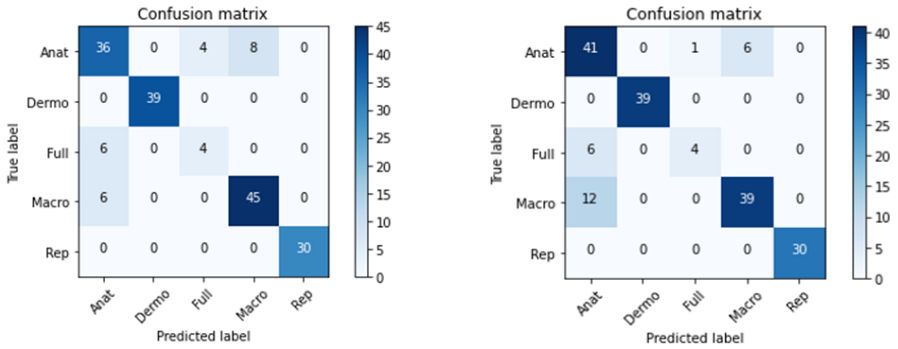
These models were also evaluated on the testing images belonging to the incremental task (task B) and on the global test set containing the testing images corresponding to both tasks (tasks A and B), to allow a later comparison with the results obtained after the implementation of the

incremental learning strategies. The corresponding accuracy results may be found in Table 5.9. Regarding the considered terminology, as explained in Section 4.3.3.3, $R_{A,B}$ corresponds to the test accuracy of task B after being trained on task A, for instance.

The confusion matrices concerning task B were also plotted and are represented in Figure 5.13. It is seen that, although the performance on this task has been lower when compared to task A, in general, both the VGG-16 and the MobileNetV2 base models were already able to predict the modalities of the images assigned to the incremental set, although they have not been trained on this task. Furthermore, the obtained accuracy with respect to this task (task B) was similar for both models. However, in the case of the VGG model it is possible to notice that more anatomic images were confused, whereas in the case of the MobileNetV2, the modality where most misclassifications were identified was the macroscopic one.

Table 5.9: Test accuracy of tasks A and B considering the selected based models. R stands for accuracy.

	$R_{A,A}$	$R_{A,B}$	$R_{A,(A+B)}$
VGG-16	0.9084	0.8652	0.9019
MobileNetV2	0.8837	0.8596	0.8801



(a) VGG model. (b) MobileNetV2 model.

Figure 5.13: Confusion matrices of the two selected models on task B.

5.2.2 Incremental Learning

After training the first task considering the selected base models, these were used as the starting point for the training of the incremental task. As already introduced, different incremental learning strategies, considering various parameters, were explored and ran for 10 iterations in order to achieve more reliable results. Also, to evaluate if training longer had an influence on the performance of the models when continuously trained in terms of forgetting, the incremental task was trained for 10, 20, and 30 epochs. As mentioned in Section 4.3.3.3, the evaluation of the employed incremental learning strategies essentially relied on their accuracy and efficiency, despite other metrics being also addressed. In what concerns the accuracy, in Tables 5.10 and 5.11, it is possible to find the achieved results, for the VGG-16 and MobileNetV2 models, respectively, and for the

different numbers of epochs. The presented results are namely in terms of the global test accuracy (considering the test images belonging to both task A and B, represented by $R_{B,(A+B)}$) after the two tasks have been trained sequentially, and of the catastrophic forgetting (given by the BWT metric). In these tables, the best performance of each strategy depending on the number of epochs is highlighted in bold. These results may further be visually analysed through Figures C.1 and C.2 contained in the Appendix.

The analysis of these tables demonstrate that, despite not being possible to completely avoid the catastrophic forgetting (as the BWT value remains negative), all of the explored incremental learning strategies allowed to reduce it. This may be verified by the increase on the BWT values when compared to the ones obtained with the Naïve strategy which works as a baseline strategy. Moreover and as a consequence, when the incremental strategies were employed, the global test accuracy also improved when compared to when the models were simply fine-tuned (Naïve strategy), which results from the fact that they allowed to preserve more information concerning the first task, as will be further discussed.

Table 5.10: VGG-16 test results for 10, 20, and 30 epochs considering different incremental learning strategies. Results averaged over 10 iterations (± SD).

	10 epochs		**20 epochs**		**30 epochs**	
	$R_{B,(A+B)}$	**BWT**	$R_{B,(A+B)}$	**BWT**	$R_{B,(A+B)}$	**BWT**
Naïve	**0.8459**±0.0025	**-0.0767**±0.0029	0.8453±0.0023	-0.0773±0.0025	0.8438±0.0032	-0.0793±0.0039
EWC100	**0.8500**±0.0025	**-0.0718**±0.0029	0.8480±0.0026	-0.0742±0.0030	0.8454±0.0031	-0.0772±0.0036
EWC50	**0.8517**±0.0039	**-0.0699**±0.0046	0.8493±0.0021	-0.0727±0.0025	0.8464±0.0041	-0.0761±0.0049
EWC1	**0.8500**±0.0028	**-0.0718**±0.0033	0.8474±00023	-0.0750±0.0027	0.8448±0.0027	-0.078±0.0033
EWC0.5	**0.8500**±0.0032	**-0.0719**±0.0038	0.8480±0.0029	-0.0741±0.0036	0.8453±0.0036	-0.0775±0.0042
AGEM50	0.8483±0.0030	-0.0738±0.0035	**0.8495**±0.0036	**-0.0724**±0.0042	0.8467±0.0034	-0.0758±0.0040
AGEM100	**0.8502**±0.0030	**-0.0716**±0.0035	0.8488±0.0030	-0.0733±0.0034	0.8476±0.0022	-0.0747±0.0025
AGEM150	**0.8522**±0.0030	**-0.0693**±0.0035	0.8489±0.0024	-0.0732±0.0029	0.8481±0.0015	-0.0741±0.0018
Replay100	0.8576±0.0046	-0.0629±0.0055	0.8597±0.0044	-0.0606±0.0050	**0.8602**±0.0056	**-0.0600**±0.0071
Replay250	0.8653±0.0036	-0.0534±0.0043	**0.8695**±0.0043	**-0.0482**±0.0050	0.8654±0.0043	-0.0537±0.0053
Replay500	0.8740±0.0055	-0.0427±0.0067	**0.8786**±0.0045	**-0.0372**±0.0056	0.8750±0.0051	-0.0419±0.0065

Table 5.11: MobileNetV2 test results for 10, 20, and 30 epochs considering different incremental learning strategies. Results averaged over 10 iterations (± SD).

	10 epochs		**20 epochs**		**30 epochs**	
	$R_{B,(A+B)}$	**BWT**	$R_{B,(A+B)}$	**BWT**	$R_{B,(A+B)}$	**BWT**
Naïve	**0.8313**±0.0049	**-0.0688**±0.0047	0.8190±0.0067	-0.0890±0.0078	0.8073±0.0039	-0.1037±0.0040
EWC100	**0.8337**±0.0072	**-0.0659**±0.0065	0.8224±0.0086	-0.0830±0.0097	0.8167±0.0096	-0.0952±0.0098
EWC50	**0.8367**±0.0065	**-0.0628**±0.0065	0.8241±0.0082	-0.0825±0.0082	0.8117±0.0113	-0.0972±0.012
EWC1	**0.8348**±0.0059	**-0.0652**±0.0057	0.8208±0.0059	-0.0862±0.0075	0.8107±0.0066	-0.1000±0.0071
EWC0.5	**0.8339**±0.0060	**-0.0655**±0.0055	0.8205±0.0074	-0.0862±0.0075	0.8158±0.0070	-0.0941±0.008
AGEM50	**0.8432**±0.0063	**-0.0567**±0.0065	0.8301±0.0078	-0.0755±0.0085	0.8205±0.0114	0.0889±0.0127
AGEM100	**0.8436**±0.0055	**-0.0563**±0.0064	0.8332±0.0059	-0.0721±0.0073	0.8265±0.0078	-0.0818±0.0095
AGEM150	**0.8453**±0.0034	**-0.0551**±0.0044	0.8358±0.0063	-0.0692±0.0078	0.8256±0.0055	-0.0828±0.0067
Replay100	**0.8516**±0.0059	**-0.0448**±0.0070	0.8485±0.0082	-0.0510±0.0095	0.8382±0.0124	-0.0643±0.0149
Replay250	**0.8588**±0.0051	**-0.0368**±0.0040	0.8534±0.0084	-0.0451±0.0090	0.8520±0.0063	-0.0481±0.0073
Replay500	**0.8604**±0.0065	**-0.0344**±0.0070	0.8559±0.0061	-0.0432±0.0073	0.8539±0.0055	-0.0456±0.0061

Regarding the influence of the number of epochs used to train the incremental task and in what concerns the VGG-16 model (Table 5.10), in the case of the Naïve, the EWC, and the AGEM strategy considering a memory size of 100 and 150, it is verified that when a higher number of epochs

was considered, the global performance of the model decreased and the catastrophic forgetting increased. Therefore, it is preferable to consider a lower number of epochs when implementing these strategies. On the other hand, in the case of the AGEM strategy with a memory buffer of 50 examples and of the Experience Replay strategy considering 250 and 500 images from the first task, it was advantageous to use an intermediate number of epochs, since when the model was trained for 20 epochs, it was possible to further reduce the forgetting, compared with the results achieved for 10 and for 30 epochs. Finally, only the Experience Replay strategy with a memory size of 100 demonstrated an improvement on the global performance when the model was trained for a larger number of epochs. In this case, even though the number of images from task A was lower, the models trained in this strategy resisted better to the catastrophic forgetting of task A for a higher number of epochs. Despite that, the longer training also resulted in a higher adaptation to the incremental images, which led to an improved global performance in the 30 epochs. Concerning the MobileNetV2 model (Table 5.11), on the other hand, it is verified that all strategies benefited from being trained for a lower number of epochs. Although the performance on task B has improved as more epochs were considered, due to the lower amount of images assigned to the incremental task, this improvement was not sufficient to compensate the increase observed on the catastrophic forgetting, leading to the decrease of the global performance of the model.

Bearing this in mind, the best performance concerning the number of epochs (the highlighted ones) was chosen for each model and strategy. Thus, in Figures 5.14 and 5.15, a comparison of the global test accuracy and the forgetting (assessed through the BWT metric), respectively, achieved by the two models and using the various incremental learning strategies may be found.

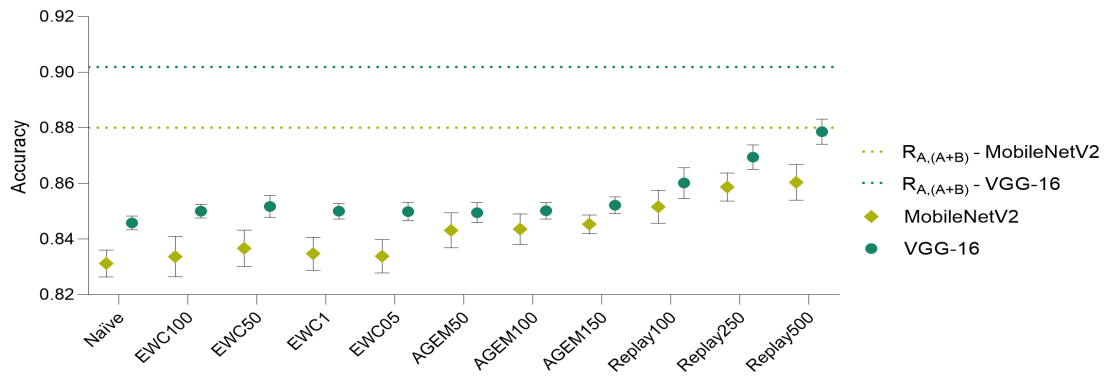


Figure 5.14: Test accuracy of both models after the incremental training.The dashed lines represent the global test accuracy after training the first task. Results averaged over 10 iterations.

Since the same base models were considered as a starting point for the incremental task, the global test accuracy after the models have been trained on the first task is the same for all strategies, and corresponds to 0.9019 and 0.8801 for the VGG-16 model and for the MobileNetV2 model, respectively (Table 5.9). Hence, in the first plot (Figure 5.14), it is possible to find two reference lines that are associated to these accuracy values. This plot also show that, after being trained incrementally, the VGG-16 model was able to achieve a higher accuracy compared to the

MobileNetV2 model. However, since the performance of the base model was already higher, the values corresponding to the difference between the accuracy achieved after the first training (represented by the dashed lines in the plot) and the accuracy achieved after the incremental one for the various strategies were considered in order to compare the performance of both models. The resulting effective values are presented in Table C.3 of the Appendix C. Taking them into account, it is possible to infer that the MobileNetV2 model outperformed the VGG-16 model since the difference between the initial accuracy and the accuracy after the training of the incremental task was lower than the difference obtained for the VGG-16 model in all of the considered strategies. Moreover and in line with this conclusion, by observing Figure 5.15, it is also possible to verify that the MobileNetV2 model surpassed the VGG-16 model in what concerns the forgetting, being able to preserve more information related to the first task. This is demonstrated by the higher BWT values obtained for all of the implemented incremental learning strategies when the MobileNetV2 model was employed, which means that the catastrophic forgetting verified with this model was lower.

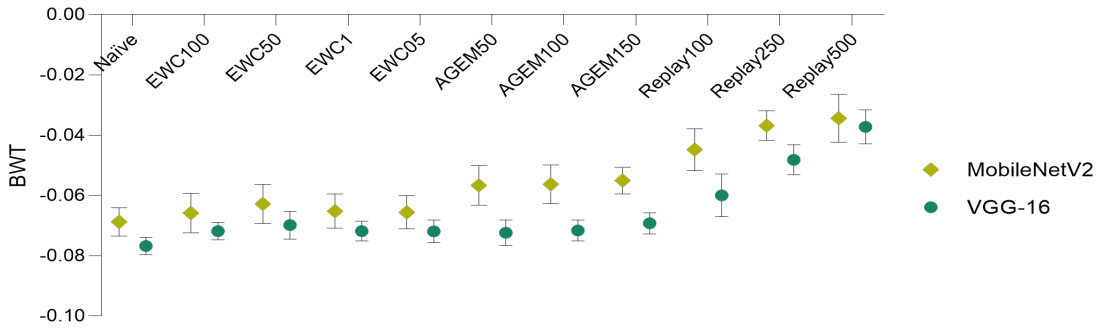


Figure 5.15: BWT of both models after the incremental training. Results averaged over 10 iterations.

Furthermore, other conclusions, namely concerning the comparison of the different incremental strategies, may be easily taken from these plots and from Tables 5.10 and 5.11. It is possible to infer that for both models, the rehearsal strategies (AGEM and Experience Replay) demonstrated to outperform the employed regularization strategy (EWC), for almost all of the considered λ values and memory sizes. This may result from the fact that these strategies imply the introduction of examples belonging to the first task during the incremental training, which allow them to better retain some of the previously acquired knowledge. Moreover, in the case of the EWC regularization strategy, the λ value that provided the best results in terms of global accuracy and forgetting corresponded to 50. On the other hand, in the case of the rehearsal strategies, it is verified that, as a higher number of patterns from the first task was considered (i.e. as the memory size increased), the performance of the models improved. Therefore, for each incremental learning strategy, a more detailed analysis was addressed taking into account the λ value (in the case of the EWC) and the memory sizes (in the case of the rehearsal strategies) that led to better outcomes. In Table 5.12, the accuracy results corresponding to the implementation of the incremental strategies using

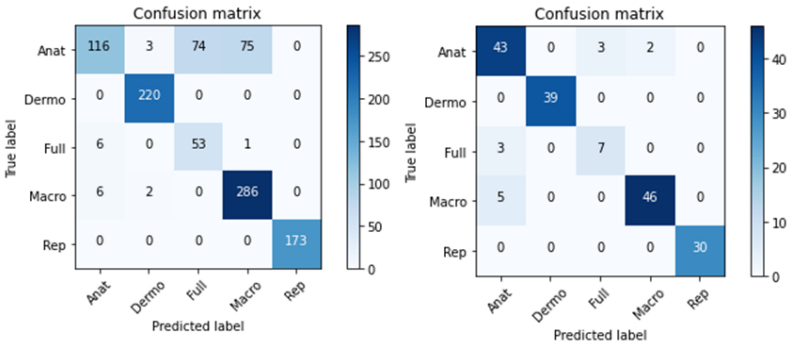
these parameters among the different tasks may be found. Nevertheless, in Table C.4, it is possible to consult these results with respect to all of the other considered strategies. These results are in line with what was previously mentioned: on the one hand, for all strategies, a decrease on the performance of the first task was verified after the incremental training has been performed, which results from the catastrophic forgetting; on the other hand, it was possible to improve the results of task B ($R_{B,B}$), when compared with the ones obtained right after the training of task A ($R_{A,B}$), since the incremental training allowed the models to fit to the incremental images.

Table 5.12: Test results in terms of accuracy concerning the best approaches for the two models. Results averaged over 10 iterations (± SD).

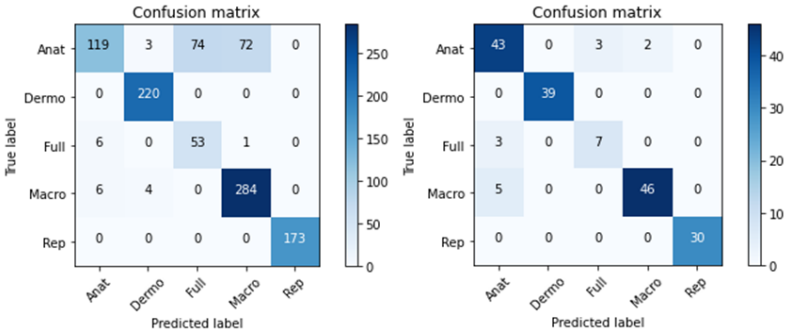
	Strategy	$R_{A,A}$	$R_{A,B}$	$R_{B,A}$	$R_{B,B}$
VGG-16	Naïve	0.9084	0.8652	0.8316±0.0029	0.9270±0.0000
	EWC50			0.8385±0.0046	0.9270±0.0000
	AGEM150			0.8391±0.0035	0.9270±0.0000
	Replay500			0.8711±0.0056	0.9213±0.0038
MobileNetV2	Naïve	0.8837	0.8596	0.8150±0.0047	0.9242±0.0100
	EWC50			0.8209±0.0065	0.9270±0.0102
	AGEM150			0.8287±0.0044	0.9404±0.0054
	Replay500			0.8494±0.0079	0.9236±0.0100

In relation to the FWT, this metric was also assessed for the two models. Since only two tasks were explored and the first training was the same for all strategies, the variables involved in the computation of this metric simply correspond to the b vector, and the $R_{A,B}$, which do not change among the various strategies. Hence, in the case of the VGG-16 model, a FTW value of 0.7303 was achieved, while in the case of the MobileNetV2 this value corresponded to 0.6910, which means that after being trained only on task A, the VGG-16 model could better perform on task B.

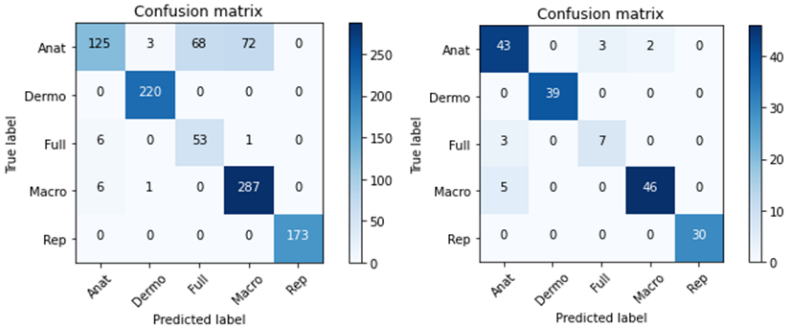
Although the accuracy is a standard metric used to evaluate incremental learning approaches, in order to assess the performance of the models in what concerns the predicted classes after they have been trained incrementally, other metrics were also computed. Once again, the evaluation of each strategy was made with respect to the parameter that provided the best results (λ value in the case of the EWC and memory sizes in the case of the rehearsal strategies). Nevertheless, in order to avoid biased results in terms of the predicted classes, an analysis concerning the results accomplished by the other parameters was also made, although the values are not presented in this document. Also, since each incremental learning strategy was ran for 10 iterations, the presented results concern a randomly selected iteration of each strategy for both models. The results concerning task A may be found in Tables C.5 and C.6 of the Appendix C. Moreover, in order to allow a visual analysis of these results, the confusion matrices corresponding to the two tasks were also plotted and are shown in Figures 5.16 and 5.17 for the VGG-16 model and the MobileNetV2, respectively.



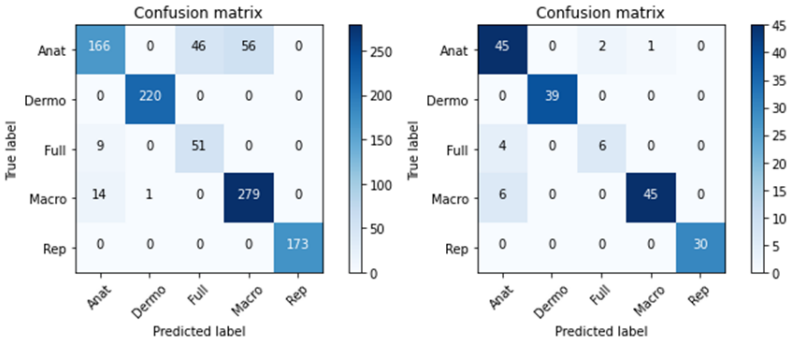
(a) Naïve strategy.



(b) EWC50 strategy.

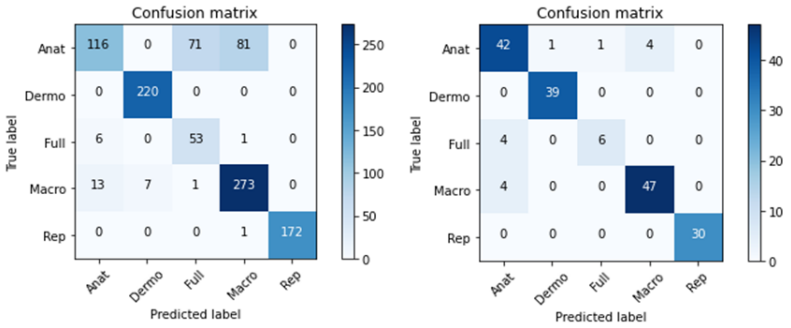


(c) AGEM150 strategy.

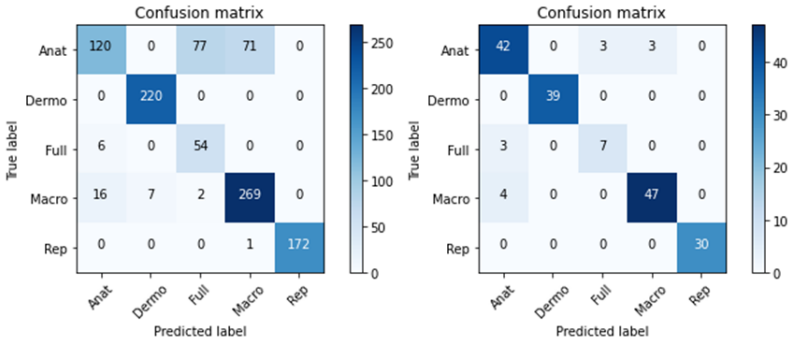


(d) Replay500 strategy.

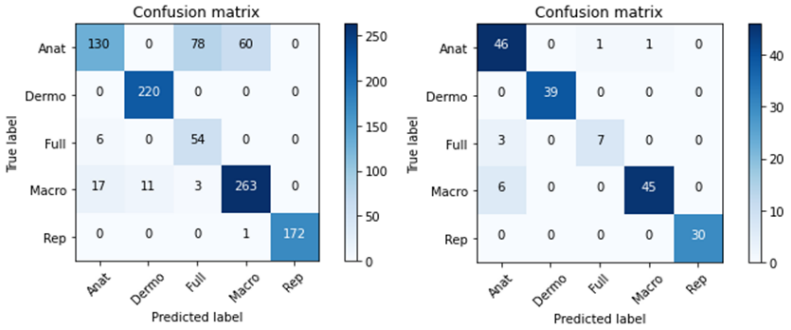
Figure 5.16: Confusion matrices achieved with the VGG-16 model for both tasks after the incremental training. The images on the left correspond to the test results of task A, and the images on the right to the results of task B.



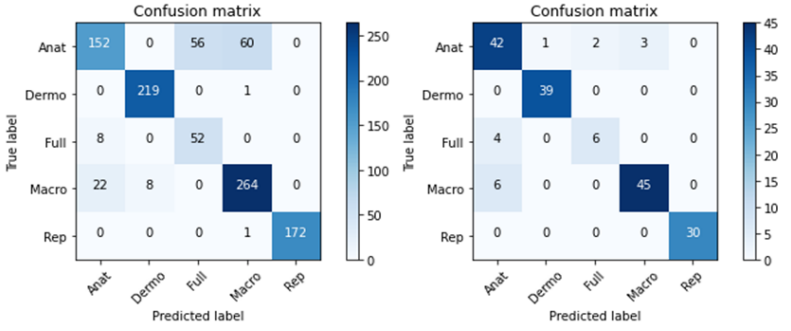
(a) Naïve strategy.



(b) EWC50 strategy.



(c) AGEM150 strategy.



(d) Replay500 strategy.

Figure 5.17: Confusion matrices achieved with the MobileNetV2 model for both tasks after the incremental training. The images on the left correspond to the test results of task A, and the images on the right to the results of task B.

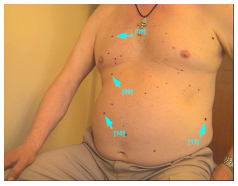
By inspecting the matrices achieved by the various strategies, it is possible to verify an identical behaviour among them and for the two considered models. In the case of task A (left images of Figures 5.16 and 5.17), confronting these plots with the ones achieved after the first training (Figure 5.9), it may be observed that the anatomic modality was the one that underwent the most changes when the incremental training was performed, being these images essentially misclassified as full-body or macroscopic images. This outcome is also demonstrated by comparing Table 5.8 with Tables C.5 and C.6 contained in the Appendix, where a steeper decrease on the F1-score metric is verified for the anatomic modality after the incremental task has been trained. This may indicate that some features of the anatomic images associated to task A are similar to the ones verified in the full-body and macroscopic images contained in the set of images assigned to the incremental task, meaning that part of the previously acquired knowledge was forgotten. Some examples of images from task A that were correctly classified after the first training but misclassified after the incremental one can be found in Figures 5.18 and 5.19, corresponding to the VGG-16 and to the MobileNetV2 model, respectively. By looking at these images, it is then possible to infer that they present a similar appearance comparing to the images that were considered in the incremental task (Section 4.3.1), despite belonging to different modalities. This may explain the alteration verified in their classification, as for instance, the anatomic modality of task B comprised images of hand and feet, whereas in the first task these images were assigned to the macroscopic class, as is the case of Figures 5.18c and 5.19c.

(a) Anatomic image classified as full-body.

(b) Anatomic image classified as macro.

(c) Macroscopic image classified as anatomic.

Figure 5.18: Examples of images from task A correctly classified after the first training but misclassified after the incremental training by the **VGG-16** model.



(a) Anatomic image classified as full-body.

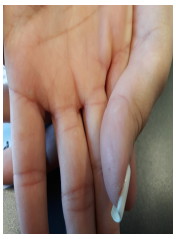
(b) Anatomic image classified as macro.

(c) Macroscopic image classified as anatomic.

Figure 5.19: Examples of images from task A correctly classified after the first training but misclassified after the incremental training by the **MobileNetV2** model.

Regarding task B, on the other hand, an improvement on its performance was verified after the models have been trained on this task. In Figures 5.20 and 5.21, some examples that demonstrate this situation are presented, corresponding to images that were previously misclassified by the models trained on the first task only but that the incremental training allowed to correctly classify. However, despite the enhanced outcomes, the shuffle between the anatomical, full-body and macroscopic modalities remained, which may even result from some aspects previously introduced, as a labeling issue, for instance.

(a) Anatomic image previously classified as full-body.



(b) Anatomic image previously classified as macro.

(c) Full-body image previously classified as anatomic.

(d) Macroscopic image previously classified as anatomic.

Figure 5.20: Examples of images from task B correctly classified after the incremental training by the **VGG-16** model.

(a) Anatomic image previously classified as full-body.

(b) Anatomic image previously classified as macro.

(c) Full-body image previously classified as anatomic.

(d) Macroscopic image previously classified as anatomic.

Figure 5.21: Examples of images from task B correctly classified after the incremental training by the **MobileNetV2** model.

Besides, as demonstrated by the corresponding matrices and by the results presented in Table 5.12, in the case of the VGG-16 model, the amount of wrongly classified images belonging to task B was the same for almost all strategies. Therefore, an analysis of the wrong predictions concerning this task was made in order to find out if the models were always failing on the same images and if these images were the same among the different strategies, which was verified. Hence, some examples of these misclassifications may be found in Figure 5.22.

(a) Anatomic image classified as full-body.

(b) Anatomic image classified as macro.

(c) Full-body image classified as anatomic.

(d) Macroscopic image classified as anatomic.

Figure 5.22: Examples of images belonging to task B misclassified by the VGG-16 model after the incremental training.

Furthermore, besides the Naïve strategy, the Cumulative strategy was also explored as a baseline strategy. As previously introduced (Section 2.7.3.1), this strategy consists of a models' retraining considering all examples from previous and new tasks. Hence, in the context of this problem, the VGG-16 and the MobileNetV2 models were trained considering both images belonging to task A and B together. The models were then tested on the images concerning the task A (R_A), task B (R_B), and on the global test set (task A + B), and the accuracy results obtained through this approach are presented in Table 5.13.

Table 5.13: Cumulative strategy results. Results averaged over 5 iterations (± SD).

	R_A	R_B	$R_{(A+B)}$
VGG-16	0.8802±0.0034	0.9224±0.0062	0.8865±0.0023
MobileNetV2	0.8617±0.0087	0.9371±0.0047	0.8742±0.0079

By comparing the results achieved with the Cumulative strategy and the ones corresponding to the incremental learning strategies (Table 5.13 and 5.12, respectively), it may be verified that the performance of task A was benefited when the models had access to all images (A+B) from the beginning, instead of being trained incrementally. This is demonstrated by the higher R_A value reached with the Cumulative strategy, compared to the $R_{B,A}$ values obtained with the incremental learning approaches. A possible explanation for this relies on the difference observed on the amount of images belonging to the two tasks, since as the first task contains more images, when the models were trained with all images together, they were not as affected by the incremental images as when the incremental training was done. On the other hand, regarding task B and the VGG-16 model, it was verified that, when all images were available (Cumulative strategy, R_B), the performance of this task was penalized in relation to an incremental training ($R_{B,B}$), except for the the Experience Replay with a memory buffer of 500 examples from the first task that achieved lower results on the incremental scenario. This exception may result from the fact that when more images from task A were considered in the incremental training, due to their higher representativeness, the incremental model could not fit so well to task B. Thus, the same aforementioned reason may be responsible for the lower performance on the cumulative scenario, since due to the fewer amount of images belonging to the incremental task, when the model was trained with all images

at the same time, it was not able to adjust so properly to it. Nevertheless, in what concerns the MobileNetV2 model, in general, the task B performance was better when the model was trained with all images together. This turns out to be in accordance with what was previously verified, as the forgetting of the MobileNetV2 was lower, which means that when trained incrementally, the model did not fit as well to the images belonging to task B, whereas when all images were trained together it was adjusted to the global domain. Moreover, comparing the overall outcomes of the models when using the incremental learning strategies ($R_{B,(A+B)}$ in Tables 5.10 and 5.11) and when trained in a cumulative scenario ($R_{(A+B)}$), it is possible to see that when the models are trained with images regarding the two tasks from the beginning, their performance improves. However, this scenario implies that all images are available at the training time, which may be unfeasible in terms of the required memory to store all examples, or even due to the computational cost involved to train them. Bearing this in mind, training models incrementally may be preferable over retraining them as new images are available.

As previously mentioned, the incremental learning strategies were also evaluated concerning their efficiency. This assessment was made both in terms of the time taken by each epoch at the learning phase, and also of the RAM memory in MegaBytes (MB) required to train the models. Although this evaluation has been made for all strategies, only the results with respect to the parameters that led to a better performance of each strategy are presented in Table 5.14.

Table 5.14: Training results in terms of efficiency concerning the best approaches for the two models. Results averaged over 10 iterations (± SD).

	Strategy	**Time/epoch(s)**	**RAM (MB)**
VGG-16	Naïve	59.38±2.66	3805.22±1360.07
	EWC50	60.98±0.64	4017.37±149.99
	AGEM150	166.01±3.44	6127.38±284.36
	Replay500	123.54±3.46	4441.09±105.13
MobileNetV2	Naïve	53.04±2.46	4437.26±226.91
	EWC50	54.88±2.23	4428.37±7.97
	AGEM150	83.91±2.97	6457.83±3.98
	Replay500	109.56±3.35	4554.59±144.08

These values demonstrate that, in terms of time, each epoch of the rehearsal strategies took longer to be trained. This results from the higher amount of considered examples, as some information concerning the first task is trained together with the incremental one. Therefore, since with respect to the VGG-16 model and in the case of the Experience Replay strategy that uses 500 examples from the first task 20 epochs were considered in the learning process, among the strategies presented in the table, this was the strategy that took the longest to be trained.

Besides that, regarding the required RAM memory, the AGEM strategy involved a higher computational cost when compared to all other strategies, being even unfeasible to be trained when a memory size higher than 150 was applied. This increase on the necessary memory may be due to the need for this strategy to estimate the averaged reference gradients using the examples contained in the memory, which implies that more information is stored at the training time.

In summary, bearing in mind the results presented in this section, it was demonstrated that:

- For both models, **with the exception of the AGEM50 strategy and the Experience Replay in the case of the VGG-16 model, it was advantageous to train for a lower number of epochs**. This happens because when the models were trained longer, although the performance on the incremental task had improved, the observed catastrophic forgetting prevailed, leading to a decrease on the global performance of the models.

- Comparing the performance of both models, **the simpler model, i.e. the MobileNetV2, was able to preserve more the previously acquired knowledge due to its lower adaptation to the incremental task**. Thus, it may be preferable over the VGG-16 model that, although have achieved a better global performance, experienced higher catastrophic forgetting.

- When the models were trained with all images from the beginning (cumulative strategy), the performance of task A improved, compared to when the models were incrementally trained. Also, concerning the task B, in the case of the VGG-16 model, the incremental training allowed it to better fit to the images contained in this task, enhancing its results. On the other hand, in the case of the MobileNetV2 model, the results of task B were better when all images were trained together, which may result from its lower capacity to adapt to the new task.

- The **rehearsal strategies (AGEM and Experience Replay) were able to achieve a better performance in terms of accuracy and forgetting when compared to the employed regularization strategy**, being the Experience Replay strategy the one that obtained the best results. It was also verified that as the memory size increased, the outcomes improved.

- Despite the performance of the rehearsal strategies being better comparing to the EWC, these strategies require that some previous images are maintained in memory to be later used in combination with the incremental set. Thus, **if it is not possible to use these images at all, taking into account that the regularization strategy was also able to achieve promising results (although lower), this strategy may be preferable over the AGEM or Experience Replay**.

- For both models, the **rehearsal strategies demonstrated to take longer to be trained when compared to the regularization strategy.** Thereby, if the training time is a relevant conditioning, the EWC strategy may be advantageous.

- With respect to the efficiency of the different strategies, the **AGEM strategy demonstrated to be the most computationally demanding one, requiring more RAM memory to be trained**.

- Comparing the two employed rehearsal strategies, although especially in the case of the VGG-16 model the Experience Replay has taken longer to be trained due to the higher number of epochs, **in what concerns the efficiency in terms of the required RAM memory**

and taking into account the obtained accuracy and forgetting results, the Experience Replay strategy may be preferable when compared to the AGEM strategy.

Chapter 6

Conclusion and Future Work

6.1 Conclusions

In the last years, the incidence of skin cancer has been increasing more and more, compromising the capacity of the medical services to respond to all patients. For this reason and thanks to the advances in medical imaging equipments, teledermatology has been an asset, ensuring an improved quality of medical care for all population. Due to the visual appearance of skin lesions, these consultations are characterized by the acquisition of images representing the patient's lesion that are stored and forwarded to a reference dermatologist, enabling the communication between the primary care units and the dermatology services. In order to facilitate the appraisal, in some cases, these images comprise a ruler next to the lesion that may be useful to later infer its size.

Nevertheless, every year, the medical records undergo an increase of around 20% to 40%, which may pose a problem for their organization since the categorization is mainly done manually, which is time-consuming, and prone to errors. In what concerns dermatology and based on the established guidelines for teledermatological consultations, it is possible to distinguish dermatological images across five different categories: anatomic, dermoscopic, full-body, macroscopic, and also clinical reports. Thus, automatic systems able to differentiate the acquired images according to these modalities and attributes (such as the presence of a ruler) may be essential to the organization of the records and consequent optimization of the teledermatological processes, as they take into account their visual characteristics. As seen in Chapter 3, different approaches have been employed in order to facilitate the access to medical images, namely through the identification of the corresponding modality. However, these systems are not specifically applied to dermatological image modalities, but to other modalities, such as CT, MRI, X-ray, and others, and for this reason, more efforts should be done in this sense.

Also, as medical data is always evolving, these systems need to be updated as new images are gathered without resorting to all of the previously seen images, as they may not be available anymore due to memory issues or the computational cost involved may be unfeasible. For these reasons, the interest for algorithms able to be incrementally trained has been growing in the last

years, allowing models to adapt to new conditions. However, training artificial intelligence systems continuously is usually prone to catastrophic forgetting which is characterized by a decrease in the performance related to the previously acquired knowledge as new data is trained. Therefore, one of the major objectives of incremental learning approaches is to face this problem, avoiding that the information already learned is lost after learning the new one.

Taking this into account, this dissertation comprised two main goals: the first one intended to compare the performance of different classification and object-detection algorithms in order to infer which was the best approach to predict whether a ruler was contained in several dermatological images; the other one consisted of the development of models able to accurately classify dermatological images according to their modality, which should employ different incremental learning strategies in order to allow their continuous training.

Concerning the first part of the work and with respect to the classification algorithms, three different models were explored: a simple CNN trained from scratch, a VGG-16 network pre-trained on the ImageNet database and fine-tuned to this binary problem, and a pre-trained VGG-16 network pre-trained on the same database and used as a feature extractor. For each model, the influence of the loss function was evaluated, having been employed a weighted cross-entropy loss function with different weights besides the standard binary cross-entropy loss. It was verified that the best configuration corresponded to the weighted cross-entropy employing a weight of 1 and of 2 to penalize the false negatives and the false positives, respectively. Moreover, data augmentation was also applied. In this case, it was demonstrated that only in the case of the fine-tuned VGG-16, there was an improvement on the algorithm's performance when more data was considered during the training process, which may result from the higher amount of parameters to be tuned in this approach. Due to the visual differences encountered in the rulers belonging to the dermoscopic class when compared to the ones contained in the other modalities, it was evaluated if this class could be compromising the results. However, this was not verified, as the performance of the three models decreased when this modality was not taken into account during the training process. Hence, considering the best configuration for each model, the simple CNN was able to achieve an accuracy of 0.917; in what concerns the fine-tuned VGG, it was possible to obtain an accuracy of 0.993, and finally, in the case of the VGG-16 used as feature extractor, an accuracy of 0.973 was reached, proving the fine-tuned VGG to be the most appropriate model to solve this binary problem.

Regarding the object-detection algorithms, on the other hand, three different detectors were considered: EfficientDet, RetinaNet, and Faster R-CNN. With the exception of the batch size and the data augmentation techniques, the considered configurations were similar to the ones used in the original implementations. In the case of EfficientDet-D0, the EfficientNet-B0 was used as backbone network; in RetinaNet two different backbone networks were explored: the ResNet-50 and the ResNet-101; and in Faster R-CNN, the ResNet-50 was also employed as the backbone network. Comparing the performance of the three detectors, the Faster R-CNN outperformed the other ones, achieving a mAP_{50} of 0.977, followed by the EfficientDet-D0 that achieved a mAP_{50} of 0.940. On the other hand, the results achieved with the RetinaNet fell short of expectations,

achieving a mAP_{50} of only 0.144 when the ResNet-50 was used and of 0.361 when the ResNet-101 was applied. Besides, in order to allow a comparison with the performance of the classification algorithms, the detectors were also evaluated considering other metrics, namely the accuracy, precision, sensitivity and specificity. Once again, Faster R-CNN provided the best results, obtaining an accuracy of 0.925.

Comparing the achieved results using both approaches (classification and object-detection), it is possible to infer that the classification algorithms surpassed the object-detection ones in the task of predicting if a ruler was contained in different dermatological images.

In what concerns the second part of the work, two models that could accurately predict the images' modalities were firstly considered: a VGG-16 and a MobileNetV2 networks pre-trained on the ImageNet database. Different regularization and rehearsal strategies were explored. Concerning the regularization strategies, the EWC strategy was implemented, whereas in the case of the rehearsal strategies, both the AGEM and the Experience Replay were addressed. These strategies were explored considering different parameters, namely the λ value in the case of the EWC strategy and the memory size in the case of the rehearsal strategies. Despite not being possible to completely avoid the catastrophic forgetting, comparing the results with the ones achieved with the Naïve strategy that works as a baseline strategy, it was demonstrated that all of the explored strategies were able to reduce it. Also, it was shown that, with respect to the rehearsal strategies, as the memory size increased, it was possible to further reduce the forgetting. Moreover, with the exception of the AGEM strategy using a memory buffer of 50 and of the Experience Replay strategy for the VGG-16 model, it was verified for all other strategies that a longer training of the incremental task led to a decrease on the models' outcomes, increasing the forgetting of the knowledge previously acquired on the first task. Furthermore, comparing the performance of the two models in terms of the ability to be incrementally trained, it was verified that the MobileNetV2 outperformed the VGG-16 model, being able to preserve more information related to the first task, as the forgetting was lower in all of the considered strategies. Comparing the outcomes of the incremental strategies to the Cumulative strategy (the other considered baseline strategy), it was verified that the performance of the first task was better when both models were trained with all images. On the other hand, in the case of the task B, whereas it has been possible to improve its performance with the incremental training employing the VGG-16 model, when using the MobileNetV2 model, its performance was better when the model had access to all images from the beginning, which may be due to its lower adaptation to the new images when performing the second training. Concerning the global performance of the models, an improvement was verified when trained with all images at the same time. However, this approach implies that images belonging to all tasks are available at the training time, which may be unfeasible. In addition, the efficiency of the incremental learning strategies was also assessed. Both rehearsal strategies demonstrated to take longer to be trained, being the Experience Replay strategy in the case of the VGG-16 model the one that took the longest, and, beyond that, the AGEM strategy was the one that required more RAM memory during training.

6.2 Future work

Regarding the first part of this work, promising results were achieved for both the classification and object-detection algorithms implemented. Nevertheless, in relation to the employed object-detectors, the results could be further improved, namely in the case of the RetinaNet detector, whose outcomes were not in line with what was expected. Therefore, this detector could be further optimized, namely with respect to the hyperparameters employed during the training phase.

Also, although this work had allowed to draw conclusions regarding the behaviour of the incremental learning approaches, there is still a lot that can be done in order to reinforce and improve the obtained results.

Firstly, as incremental learning strategies intend to enable the models to adapt to new conditions, in the case of a new-instances scenario which was the one here involved (Section 2.7.2), they expect that new information is included in the incremental tasks, in order to represent the concept-drift. Therefore, since within the scope of this work, the images considered in the incremental task were part of the same original dataset and this selection was visually made, it could be interesting to use different images on this task, in order to verify if the main conclusions still hold.

Due to the limited amount of data, it was not possible to investigate the performance of the incremental learning strategies in more than one incremental task. Hence, to corroborate the achieved results, more data should be considered allowing to include another incremental task. This task could take into account images with different skin tones (as the ones used in this work were essentially from a fair-skinned population) or even the dataset could be divided among the different tasks according to the images' acquisition protocols, for instance.

Concerning the AGEM strategy, due to the involved computational cost, the higher memory size that could be explored was of 150. However, to allow a better comparison of the rehearsal strategies' performance in terms of accuracy and forgetting, in a future work, the same memory sizes should be considered for both strategies.

Appendix A

State-of-the-Art of Medical Imaging Modality Classification

A.1 Hand-crafted based approaches

Table A.1: Hand-crafted approaches for modality classification

Reference	Types of features	Features or methods	Classification strategy	Best accuracy
Kalpathy *et al.*[83]	Visual	Histogram and texture features	Neural network based	>95%
Song *et al.*[180]	Visual	Edge Histogram, Tamura and Gabor	SVM	60.01%
Khachane *et al.*[181]	Visual	Mean, standard deviation and image contrast	Fuzzy rule-based system	84.00%
Arias *et al.*[182]	Visual	BoVW-SIFT, BoC, CEDD, FCTH, FCH	BNC-AODE	69.21%
Cao *et al.* [184]	Visual	LPB, edge and color histograms, SIFT, and others	Multiclass SVM	69.70%
Valavanis *et al.*[185]	Visual	PHoW and QBoC	Multiclass SVM	84.01%
Markonis *et al.*[186]	Visual	BoVW-SIFT and GIFT	k-NN	86.90%

Table A.1: Hand-crafted approaches for modality classification *(cont.)*

Reference	**Types of features**	**Features or methods**	**Classification strategy**	**Best accuracy**
Dimitrovski *et al.*[82]	Visual and Textual	Visual: LBP, FCTH, CEDD, and SIFT; Textual: BoW with TF-IDF weighting	Multiclass SVM	87.10%
Kitanovski *et al.*[187]	Visual and Textual	Visual: BoVW-SIFT; Textual: TF-IDF weighting	SVM with χ^2 kernel	78.04%
Pelka *et al.*[189]	Visual and Textual	Visual: Tamura, Gabor, FCH, BoK-SIFT, and others; Textual: BoW	SVM and Random Forest	67.60%
Wu *et al.*[190]	Visual and Textual	Visual: SIFT, LBP, Gabor, and Tamura; Textual: VSM with TF-IDF weighting	Multiclass SVM	95.15%
Gál *et al.*[191]	Visual and Textual	Visual: BoVW-SIFT, colour histogram, mean of pixels and others; Textual: Image caption, meta-data	SVM	86.03%
Csurka *et al.*[81]	Visual and Textual	Visual: BoVW and Fisher vector; Textual: BoW	Logistic regression with Laplace Prior	94.40%

A.2 Deep neural network based approaches

Table A.2: Deep neural networks based approaches for modality classification

Reference	Database	Base Architecture	Classification strategy	Best Accuracy
Kumar *et al.* [192]	ImageCLEF 2016	AlexNet and GoogleNet	Softmax and multiclass SVM	82.48% (top 1) and 96.59% (top 5)
Kumar *et al.* [193]	ImageCLEF 2016	AlexNet	Multiclass SVM	77.55%
Semedo *et al.* [194]	ImageCLEF 2016	VGG and VGG with PReLU	Softmax	65.31%
Yu *et al.* [19]	ImageCLEFmed 2013	CNNs from scratch	Softmax	74.90%
Yu *et al.* [196]	ImageCLEF2015 and Image-CLEF2016	VGGNet; ResNet and CNN from scratch	Softmax	81.86% (Image-CLEF2016)
Hassan *et al.* [79]	ImageCLEF2012	ResNet50	LDA	87.91%
Singh *et al.* [78]	Open-i Biomedical Image Search Engine	VGG-16, ResNet-50, Inception-v3, and 4 others	Logistic regression	—
Khan and Yong[77]	ImageCLEF2012	CNN from scratch	Softmax	81.2%
Zhang *et al.* [197]	ImageCLEF2016	ResNet50	Softmax	86.58%

Appendix B

Ruler inference

B.1 Object-detection algorithms

Figure B.1: Visual results of the object-detection algorithms - Examples of EfficientDet's True Positives.

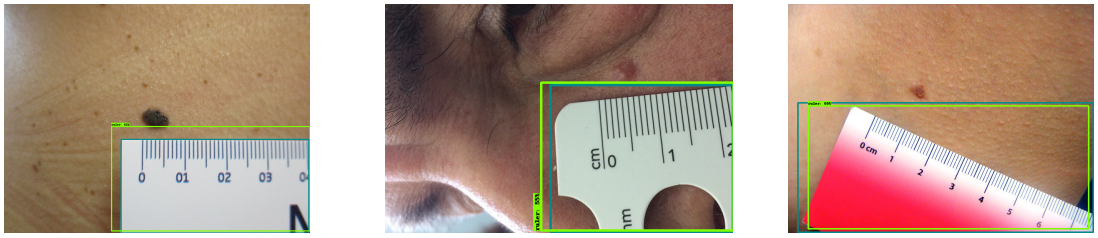


Figure B.2: Visual results of the object-detection algorithms - Examples of RetinaNet's True Positives.

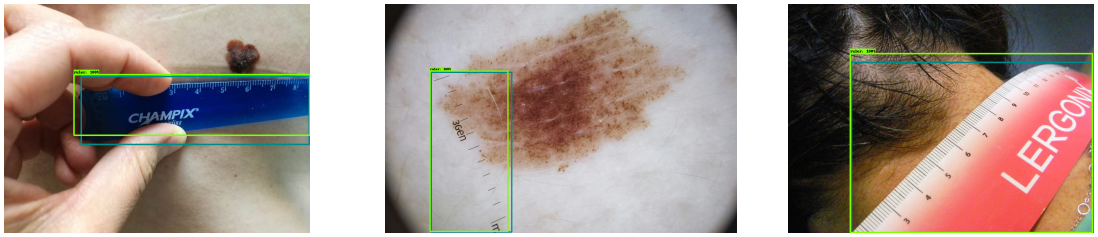


Figure B.3: Visual results of the object-detection algorithms - Examples of Faster R-CNN's True Positives.

Appendix C

Image Modality Classification

C.1 Base model selection

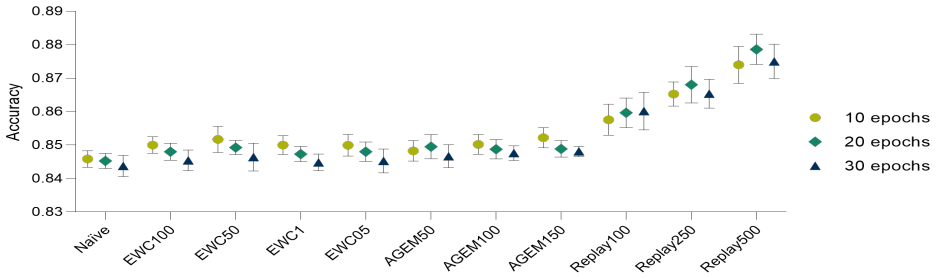
Table C.1: Results of the non-selected VGG-16 base models.

Model	Modality	Accuracy	Precision	Recall	F1-score
VGG-16	Anatomic	0.8985	0.9073	0.6940	0.7865
	Dermoscopic		0.9778	1.0000	0.9888
	Full-body		0.6098	0.8333	0.7042
	Macroscopic		0.8576	0.9626	0.9071
	Clinical reports		1.0000	1.0000	1.0000
	Macro average	—	0.8705	0.8980	0.8773
	Weighted average	—	0.9064	0.8985	0.8968
VGG-16	Anatomic	0.8985	0.9029	0.6940	0.7848
	Dermoscopic		0.9821	1.0000	0.9910
	Full-body		0.6282	0.8167	0.7101
	Macroscopic		0.8503	0.9660	0.9045
	Clinical reports		1.0000	1.0000	1.0000
	Macro average	—	0.8727	0.8953	0.8781
	Weighted average	—	0.9052	0.8985	0.8964

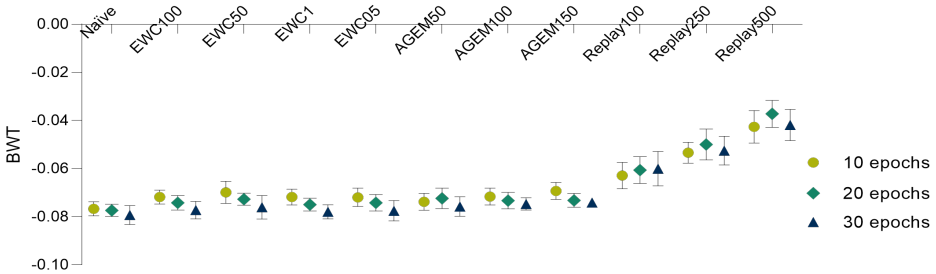
Table C.2: Results of the non-selected MobileNetV2 base models.

Model	**Modality**	**Accuracy**	**Precision**	**Recall**	**F1-score**
MobileNetV2	Anatomic	0.8818	0.8047	0.7687	0.7863
	Dermoscopic		0.9733	0.9955	0.9843
	Full-body		0.6515	0.7167	0.6825
	Macroscopic		0.8615	0.8673	0.8644
	Clinical reports		1.0000	0.9942	0.9971
	Macro average	—	0.8582	0.8685	0.8629
	Weighted average	—	0.8819	0.8818	0.8816
MobileNetV2	Anatomic	0.8768	0.8240	0.7164	0.7665
	Dermoscopic		0.9777	0.9955	0.9865
	Full-body		0.5465	0.7833	0.6438
	Macroscopic		0.8667	0.8844	0.8754
	Clinical reports		1.0000	0.9942	0.9971
	Macro average	—	0.8430	0.8748	0.8539
	Weighted average	—	0.8833	0.8768	0.8778

C.2 Incremental Learning

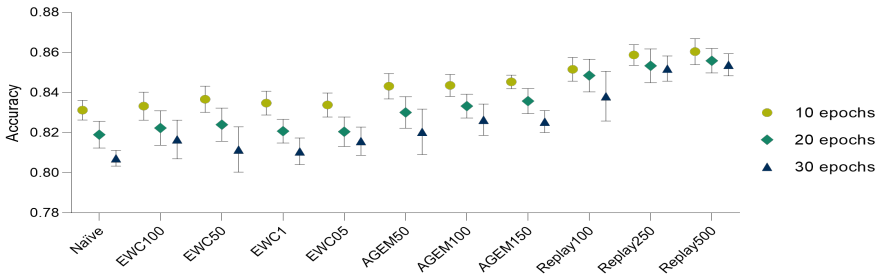


(a) Global test accuracy.

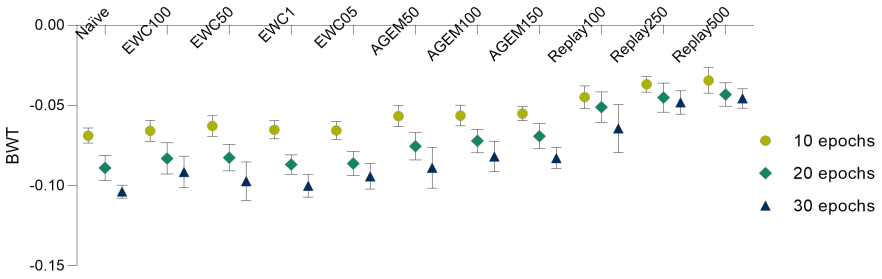


(b) Backward Transfer.

Figure C.1: Test results of the **VGG-16** model after the incremental learning in terms of the considered number of epochs.



(a) Global test accuracy.



(b) Backward Transfer.

Figure C.2: Test results of the **MobileNetV2** model after the incremental learning in terms of the considered number of epochs.

Table C.3: Effective values corresponding to the difference between the global test accuracy after training on task A and after training on task B.

	Naïve	EWC100	EWC50	EWC1	EWC05	AGEM50	AGEM100	AGEM150	Replay100	Replay250	Replay500
VGG-16	0.0560	0.0519	0.0502	0.0519	0.0519	0.0524	0.0517	0.0497	0.0417	0.0335	0.0233
MobileNetV2	0.0488	0.0464	0.0434	0.0453	0.0462	0.0369	0.0365	0.0348	0.0285	0.0213	0.0197

Table C.4: Accuracy results over the various tasks for both models. Results averaged over 10 iterations ($\pm$ SD).

	Strategy	$R_{A,A}$	$R_{A,B}$	$R_{B,A}$	$R_{B,B}$
	Naïve			0.8316±0.0029	0.9270±0.0000
	EWC100			0.8366±0.0029	0.9270±0.0000
	EWC50			0.8385±0.0046	0.9270±0.0000
	EWC1			0.8365±0.0033	0.9270±0.0000
	EWC05			0.8365±0.0038	0.9270±0.0000
VGG-16	AGEM50	0.9084	0.8652	0.8360±0.0043	0.9270±0.0000
	AGEM100			0.8367±0.0035	0.9270±0.0000
	AGEM150			0.8391±0.0035	0.9270±0.0000
	Replay100			0.8484±0.0071	0.9275±0.0062
	Replay250			0.8602±0.0049	0.9224±0.0024
	Replay500			0.8711±0.0056	0.9213±0.0038
	Naïve			0.8150±0.0047	0.9242±0.0100
	EWC100			0.8178±0.0065	0.9213±0.0116
	EWC50			0.8209±0.0065	0.9270±0.0103
	EWC1			0.8185±0.0057	0.9275±0.0111
	EWC05			0.8181±0.0055	0.9236±0.0110
MobileNetV2	AGEM50	0.8837	0.8596	0.8271±0.0066	0.9348±0.0093
	AGEM100			0.8275±0.0064	0.9354±0.0055
	AGEM150			0.8287±0.0044	0.9404±0.0054
	Replay100			0.8389±0.0070	0.9242±0.0100
	Replay250			0.8469±0.0049	0.9264±0.0120
	Replay500			0.8494±0.0079	0.9236±0.0100

Table C.5: Results achieved with the **VGG-16** model for a random iteration after the incremental learning tested on task A.

Strategy	**Modality**	**Accuracy**	**Precision**	**Recall**	**F1-score**
Naïve	Anatomic	0.8355	0.9062	0.4328	0.5859
	Dermoscopic		0.9778	1.0000	0.9888
	Full-body		0.4173	0.8833	0.5668
	Macroscopic		0.7901	0.9728	0.8720
	Clinical reports		1.0000	1.0000	1.0000
	Macro average	—	0.8183	0.8578	0.8027
	Weighted average	—	0.8752	0.8355	0.8255
EWC50	Anatomic	0.8365	0.9084	0.4440	0.5965
	Dermoscopic		0.9692	1.0000	0.9843
	Full-body		0.4173	0.8833	0.5668
	Macroscopic		0.7955	0.9660	0.8725
	Clinical reports		1.0000	1.0000	1.0000
	Macro average	—	0.8181	0.8587	0.8040
	Weighted average	—	0.8755	0.8365	0.8275
AGEM150	Anatomic	0.8453	0.9124	0.4664	0.6173
	Dermoscopic		0.9821	1.0000	0.9910
	Full-body		0.4380	0.8833	0.5856
	Macroscopic		0.7972	0.9762	0.8777
	Clinical reports		1.0000	1.0000	1.0000
	Macro average	—	0.8260	0.8652	0.8143
	Weighted average	—	0.8810	0.8453	0.8371
Replay500	Anatomic	0.8759	0.8783	0.6194	0.7265
	Dermoscopic		0.9955	1.0000	0.9977
	Full-body		0.5258	0.8500	0.6497
	Macroscopic		0.8328	0.9490	0.8871
	Clinical reports		1.0000	1.0000	1.0000
	Macro average	—	0.8465	0.8837	0.8522
	Weighted average	—	0.8904	0.8759	0.8739

Table C.6: Results achieved with the **MobileNetV2** model for a random iteration after the incremental learning tested on task A.

Strategy	**Modality**	**Accuracy**	**Precision**	**Recall**	**F1-score**
Naïve	Anatomic	0.8217	0.8593	0.4328	0.5757
	Dermoscopic		0.9692	1.0000	0.9843
	Full-body		0.4240	0.8833	0.5730
	Macroscopic		0.7669	0.9286	0.8400
	Clinical reports		1.000	0.9942	0.9971
	Macro average	—	0.8039	0.8478	0.7940
	Weighted average	—	0.8546	0.8217	0.8125
EWC50	Anatomic	0.8227	0.8451	0.4478	0.5854
	Dermoscopic		0.9692	1.0000	0.9843
	Full-body		0.4060	0.9000	0.5596
	Macroscopic		0.7889	0.9150	0.8472
	Clinical reports		1.0000	0.9942	0.9971
	Macro average	—	0.8018	0.8514	0.7947
	Weighted average	—	0.8561	0.8227	0.8164
AGEM150	Anatomic	0.8266	0.8497	0.4851	0.6176
	Dermoscopic		0.9524	1.0000	0.9756
	Full-body		0.4000	0.9000	0.5538
	Macroscopic		0.8117	0.8946	0.8511
	Clinical reports		1.0000	0.9942	0.9971
	Macro average	—	0.8028	0.8548	0.7991
	Weighted average	—	0.8600	0.8266	0.8238
Replay500	Anatomic	0.8463	0.8352	0.5672	0.6756
	Dermoscopic		0.9648	0.9955	0.9799
	Full-body		0.4815	0.8667	0.6190
	Macroscopic		0.8098	0.8980	0.8516
	Clinical reports		1.0000	0.9942	0.9971
	Macro average	—	0.8182	0.8643	0.8246
	Weighted average	—	0.8631	0.8463	0.8440

Bibliography

[1] J. K. Cullen, J. L. Simmons, P. G. Parsons, and G. M. Boyle, "Topical treatments for skin cancer," *Advanced drug delivery reviews*, vol. 153, pp. 54–64, 2020.

[2] Z. Apalla, D. Nashan, R. B. Weller, and X. Castellsagué, "Skin cancer: Epidemiology, disease burden, pathophysiology, diagnosis, and therapeutic approaches," *Dermatology and therapy*, vol. 7, no. 1, pp. 5–19, 2017.

[3] V. Samarasinghe and V. Madan, "Nonmelanoma skin cancer," *Journal of cutaneous and aesthetic surgery*, vol. 5, no. 1, p. 3, 2012.

[4] World Health Organization, *Estimated number of new cases in 2020*, `https://gco.iarc.fr/today/online-analysis-table?v=2020&mode=cancer&mode_population=continents&population=900&populations=900&key=asr&sex=0&cancer=39&type=0&statistic=5&prevalence=0&population_group=0&ages_group%5B%5D=0&ages_group%5B%5D=17&group_cancer=0&include_nmsc=1&include_nmsc_other=0`, Accessed: 2021-02-01.

[5] M. Trakatelli, C. Ulrich, V. Del Marmol, S. Euvrard, E. Stockfleth, and D. Abeni, "Epidemiology of nonmelanoma skin cancer (nmsc) in europe: Accurate and comparable data are needed for effective public health monitoring and interventions," *British journal of dermatology*, vol. 156, pp. 1–7, 2007.

[6] J. Ferlay, M. Colombet, I. Soerjomataram, C. Mathers, D. Parkin, M. Piñeros, A. Znaor, and F. Bray, "Estimating the global cancer incidence and mortality in 2018: Globocan sources and methods," *International journal of cancer*, vol. 144, no. 8, pp. 1941–1953, 2019.

[7] U. Leiter, U. Keim, and C. Garbe, "Epidemiology of skin cancer: Update," *Sunlight, Vitamin D and Skin Cancer*, p. 123,

[8] V. Nikolaou and A. Stratigos, "Emerging trends in the epidemiology of melanoma," *British journal of dermatology*, vol. 170, no. 1, pp. 11–19, 2014.

[9] American Cancer Society, "Cancer facts & figures 2020," *CA: A Cancer Journal for Clinicians*, 2020.

[10] H. M. Gloster Jr and D. G. Brodland, "The epidemiology of skin cancer," *Dermatologic Surgery*, vol. 22, no. 3, pp. 217–226, 1996.

[11] H. Kittler, H. Pehamberger, K. Wolff, and M. Binder, "Diagnostic accuracy of dermoscopy," *The lancet oncology*, vol. 3, no. 3, pp. 159–165, 2002.

[12] E. Errichetti and G. Stinco, "Dermoscopy in general dermatology: A practical overview," *Dermatology and therapy*, vol. 6, no. 4, pp. 471–507, 2016.

[13] D. Schadendorf, A. C. van Akkooi, C. Berking, K. G. Griewank, R. Gutzmer, A. Hauschild, A. Stang, A. Roesch, and S. Ugurel, "Melanoma," *The Lancet*, vol. 392, no. 10151, pp. 971–984, 2018.

[14] A. Rezvantalab, H. Safigholi, and S. Karimijeshni, "Dermatologist level dermoscopy skin cancer classification using different deep learning convolutional neural networks algorithms," *arXiv preprint arXiv:1810.10348*, 2018.

[15] A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, and S. Thrun, "Dermatologist-level classification of skin cancer with deep neural networks," *nature*, vol. 542, no. 7639, pp. 115–118, 2017.

[16] T. J. Brinker, A. Hekler, A. H. Enk, C. Berking, S. Haferkamp, A. Hauschild, M. Weichenthal, J. Klode, D. Schadendorf, T. Holland-Letz, *et al.*, "Deep neural networks are superior to dermatologists in melanoma image classification," *European Journal of Cancer*, vol. 119, pp. 11–17, 2019.

[17] H. A. Haenssle, C. Fink, R. Schneiderbauer, F. Toberer, T. Buhl, A. Blum, A. Kalloo, A. B. H. Hassen, L. Thomas, A. Enk, *et al.*, "Man against machine: Diagnostic performance of a deep learning convolutional neural network for dermoscopic melanoma recognition in comparison to 58 dermatologists," *Annals of Oncology*, vol. 29, no. 8, pp. 1836–1842, 2018.

[18] M. Abedini, N. C. Codella, J. H. Connell, R. Garnavi, M. Merler, S. Pankanti, J. R. Smith, and T. Syeda-Mahmood, "A generalized framework for medical image classification and recognition," *IBM Journal of Research and Development*, vol. 59, no. 2/3, pp. 1–1, 2015.

[19] Y. Yu, H. Lin, Q. Yu, J. Meng, Z. Zhao, Y. Li, and L. Zuo, "Modality classification for medical images using multiple deep convolutional neural networks," *J. Comput. Inf. Syst*, vol. 11, no. 15, pp. 5403–5413, 2015.

[20] T. Nedelcu, M. Vasconcelos, and A. Carreiro, "Multi-dataset training for skin lesion classification on multimodal and multitask deep learning,"

[21] Direção-Geral da Saúde, *Telerrastreio dermatológico*, `https://www.dgs.pt/directrizes-da-dgs/normas-e-circulares-normativas/norma-n-0052014-de-08042014-pdf.aspx`, Accessed: 2020-12-16.

[22] D. Roy, P. Panda, and K. Roy, "Tree-cnn: A hierarchical deep convolutional neural network for incremental learning," *Neural Networks*, vol. 121, pp. 148–160, 2020.

[23] Fraunhofer AICOS, *Derm.AI - Usage of Artificial Intelligence to Power Teledermatological Screening*, `http://dermai.projects.fraunhofer.pt/details.html`, Accessed: 2021-03-02.

[24] International Agency for Research on Cancer WHO, *Portugal*, `https://gco.iarc.fr/today/data/factsheets/populations/620-portugal-fact-sheets.pdf`, Accessed: 2020-12-06.

[25] E. Perera, N. Gnaneswaran, C. Staines, A. K. Win, and R. Sinclair, "Incidence and prevalence of non-melanoma skin cancer in a ustralia: A systematic review," *Australasian Journal of Dermatology*, vol. 56, no. 4, pp. 258–267, 2015.

[26] T. L. Diepgen and V. Mahler, "The epidemiology of skin cancer," *British Journal of Dermatology*, vol. 146, pp. 1–6, 2002.

[27] N. Telfer, G. Colver, and C. Morton, "Guidelines for the management of basal cell carcinoma," *British journal of Dermatology*, vol. 159, no. 1, pp. 35–48, 2008.

[28] F. J. Bath-Hextall, W. Perkins, J. Bong, and H. C. Williams, "Interventions for basal cell carcinoma of the skin," *Cochrane database of systematic reviews*, no. 1, 2007.

[29] Skin Cancer Foundation, *Skin Cancer*, `https://www.skincancer.org/skin-cancer-information/`, Accessed: 2020-12-12.

[30] A. N. Crowson, "Basal cell carcinoma: Biology, morphology and clinical implications," *Modern pathology*, vol. 19, no. 2, S127–S147, 2006.

[31] C. Wong, R. Strange, and J. Lear, "Basal cell carcinoma," *Bmj*, vol. 327, no. 7418, pp. 794–798, 2003.

[32] V. Madan, J. T. Lear, and R.-M. Szeimies, "Non-melanoma skin cancer," *The lancet*, vol. 375, no. 9715, pp. 673–685, 2010.

[33] A. I. Rubin, E. H. Chen, and D. Ratner, "Basal-cell carcinoma," *New England Journal of Medicine*, vol. 353, no. 21, pp. 2262–2269, 2005.

[34] J. Roewert-Huber, B. Lange-Asschenfeldt, E. Stockfleth, and H. Kerl, "Epidemiology and aetiology of basal cell carcinoma," *British Journal of Dermatology*, vol. 157, pp. 47–51, 2007.

[35] DermNet New Zealand Trust, *DermNet NZ - All about the skin*, `https://www.dermnetnz.org/`, Accessed: 2020-12-23.

[36] S. J. Miller, "Biology of basal cell carcinoma (part i)," *Journal of the American Academy of Dermatology*, vol. 24, no. 1, pp. 1–13, 1991.

[37] V. Bartoš, D. Pokorn, O. Zacharová, P. Haluska, J. Doboszová, M. Kullová, K. Adamicová, M. Péč, and J. Péč, "Recurrent basal cell carcinoma: A clinicopathological study and evaluation of histomorphological findings in primary and recurrent lesions," *Acta Dermatoven APA*, vol. 20, no. 2, 2011.

[38] American Cancer Society, *What Are Basal and Squamous Cell Skin Cancers?* https://www.cancer.org/cancer/basal-and-squamous-cell-skin-cancer/about/what-is-basal-and-squamous-cell.html, Accessed: 2020-12-13.

[39] R. Marks, "Squamous cell carcinoma," *The Lancet*, vol. 347, no. 9003, pp. 735–738, 1996.

[40] J. M. Firnhaber, "Diagnosis and treatment of basal cell and squamous cell carcinoma," *American family physician*, vol. 86, no. 2, pp. 161–168, 2012.

[41] M. Alam and D. Ratner, "Cutaneous squamous-cell carcinoma," *New England Journal of Medicine*, vol. 344, no. 13, pp. 975–983, 2001.

[42] D. S. Cassarino, D. P. DeRienzo, and R. J. Barr, "Cutaneous squamous cell carcinoma: A comprehensive clinicopathologic classification: Part two," *Journal of cutaneous pathology*, vol. 33, no. 4, pp. 261–279, 2006.

[43] S. K. T. Que, F. O. Zwald, and C. D. Schmults, "Cutaneous squamous cell carcinoma: Incidence, risk factors, diagnosis, and staging," *Journal of the American Academy of Dermatology*, vol. 78, no. 2, pp. 237–247, 2018.

[44] R. E. Kwa, K. Campana, and R. L. Moy, "Biology of cutaneous squamous cell carcinoma," *Journal of the American Academy of Dermatology*, vol. 26, no. 1, pp. 1–26, 1992.

[45] D. L. Cummins, J. M. Cummins, H. Pantle, M. A. Silverman, A. L. Leonard, and A. Chanmugam, "Cutaneous malignant melanoma," in *Mayo clinic proceedings*, Elsevier, vol. 81, 2006, pp. 500–507.

[46] B. H. Porras and C. J. Cockerell, "Cutaneous malignant melanoma: Classification and clinical diagnosis.," in *Seminars in cutaneous medicine and surgery*, vol. 16, 1997, pp. 88–96.

[47] E. de Vries and J. W. Coebergh, "Cutaneous malignant melanoma in europe," *European Journal of Cancer*, vol. 40, no. 16, pp. 2355–2366, 2004.

[48] SkinVision, *How do common skin lesions look?* https://www.skinvision.com/articles/how-do-common-skin-lesions-look-with-pictures/, Accessed: 2021-01-20.

[49] C. Hafner and T. Vogt, "Seborrheic keratosis," *JDDG: Journal der Deutschen Dermatologischen Gesellschaft*, vol. 6, no. 8, pp. 664–677, 2008.

[50] R. P. Braun, H. S. Rabinovitz, J. Krischer, J. Kreusch, M. Oliviero, L. Naldi, A. W. Kopf, and J. H. Saurat, "Dermoscopy of pigmented seborrheic keratosis: A morphological study," *Archives of dermatology*, vol. 138, no. 12, pp. 1556–1560, 2002.

[51] M. R. Roh, P. Eliades, S. Gupta, and H. Tsao, "Genetics of melanocytic nevi," *Pigment cell & melanoma research*, vol. 28, no. 6, pp. 661–672, 2015.

[52] H. Tsao, C. Bevona, W. Goggins, and T. Quinn, "The transformation rate of moles (melanocytic nevi) into cutaneous melanoma: A population-based estimate," *Archives of dermatology*, vol. 139, no. 3, pp. 282–288, 2003.

[53] C. Praetorius, R. A. Sturm, and E. Steingrimsson, "Sun-induced freckling: Ephelides and solar lentigines," *Pigment Cell & Melanoma Research*, vol. 27, no. 3, pp. 339–350, 2014.

[54] J.-P. Ortonne, A. G. Pandya, H. Lui, and D. Hexsel, "Treatment of solar lentigines," *Journal of the American Academy of Dermatology*, vol. 54, no. 5, S262–S271, 2006.

[55] P. Zaballos, S. Puig, A. Llambrich, and J. Malvehy, "Dermoscopy of dermatofibromas: A prospective morphological study of 412 cases," *Archives of dermatology*, vol. 144, no. 1, pp. 75–83, 2008.

[56] E. Fagundo, C. Rodrıguez-Garcıa, C. Rodrıguez, S. González, R. Sánchez, and A. Jiménez, "Analysis of phenotypic characteristics and exposure to uv radiation in a group of patients with cutaneous melanoma," *Actas Dermo-Sifiliográficas (English Edition)*, vol. 102, no. 8, pp. 599–604, 2011.

[57] R. Gordon, "Skin cancer: An overview of epidemiology and risk factors," in *Seminars in oncology nursing*, Elsevier, vol. 29, 2013, pp. 160–169.

[58] H. K. Koh, A. C. Geller, D. R. Miller, T. A. Grossbart, and R. A. Lew, "Prevention and early detection strategies for melanoma and skin cancer: Current status," *Archives of dermatology*, vol. 132, no. 4, pp. 436–443, 1996.

[59] S.-M. Hwang, H.-C. Pan, M.-K. Hwang, M.-W. Kim, and J.-S. Lee, "Malignant skin tumor misdiagnosed as a benign skin lesion," *Archives of craniofacial surgery*, vol. 17, no. 2, p. 86, 2016.

[60] G. Argenziano, H. P. Soyer, S. Chimenti, R. Talamini, R. Corona, F. Sera, M. Binder, L. Cerroni, G. De Rosa, G. Ferrara, *et al.*, "Dermoscopy of pigmented skin lesions: Results of a consensus meeting via the internet," *Journal of the American Academy of Dermatology*, vol. 48, no. 5, pp. 679–693, 2003.

[61] F. Nachbar, W. Stolz, T. Merkle, A. B. Cognetta, T. Vogt, M. Landthaler, P. Bilek, O. Braun-Falco, and G. Plewig, "The abcd rule of dermatoscopy: High prospective value in the diagnosis of doubtful melanocytic skin lesions," *Journal of the American Academy of Dermatology*, vol. 30, no. 4, pp. 551–559, 1994.

[62] N. R. Abbasi, H. M. Shaw, D. S. Rigel, R. J. Friedman, W. H. McCarthy, I. Osman, A. W. Kopf, and D. Polsky, "Early diagnosis of cutaneous melanoma: Revisiting the abcd criteria," *Jama*, vol. 292, no. 22, pp. 2771–2776, 2004.

[63] H. P. Soyer, G. Argenziano, V. RUOCCO, and S. CHIMENTI, "Dermoscopy of pigmented skin lesions*(part ii)," *European Journal of Dermatology*, vol. 11, no. 5, pp. 483–98, 2001.

[64] J. Kawahara, S. Daneshvar, G. Argenziano, and G. Hamarneh, "Seven-point checklist and skin lesion classification using multitask multimodal neural nets," *IEEE journal of biomedical and health informatics*, vol. 23, no. 2, pp. 538–546, 2018.

[65] G. Fabbrocini, V. De Vita, S. Cacciapuoti, G. Di Leo, C. Liguori, A. Paolillo, A. Pietrosanto, and P. Sommella, "Automatic diagnosis of melanoma based on the 7-point checklist," in *Computer Vision Techniques for the Diagnosis of Skin Cancer*, Springer, 2014, pp. 71–107.

[66] C. Dolianitis, J. Kelly, R. Wolfe, and P. Simpson, "Comparative performance of 4 dermoscopic algorithms by nonexperts for the diagnosis of melanocytic lesions," *Archives of dermatology*, vol. 141, no. 8, pp. 1008–1014, 2005.

[67] R. B. Oliveira, J. P. Papa, A. S. Pereira, and J. M. R. Tavares, "Computational methods for pigmented skin lesion classification in images: Review and future trends," *Neural Computing and Applications*, vol. 29, no. 3, pp. 613–636, 2018.

[68] E. M. Warshaw, Y. J. Hillman, N. L. Greer, E. M. Hagel, R. MacDonald, I. R. Rutks, and T. J. Wilt, "Teledermatology for diagnosis and management of skin conditions: A systematic review," *Journal of the American Academy of Dermatology*, vol. 64, no. 4, pp. 759–772, 2011.

[69] D. Eedy and R. Wootton, "Teledermatology: A review," *British Journal of Dermatology*, vol. 144, no. 4, pp. 696–707, 2001.

[70] D. A. Perednia and N. Brown, "Teledermatology: One application of telemedicine.," *Bulletin of the Medical Library Association*, vol. 83, no. 1, p. 42, 1995.

[71] J. D. Whited, "Teledermatology research review," *International journal of dermatology*, vol. 45, no. 3, pp. 220–229, 2006.

[72] American Telemedicine Association, *Practice guidelines for dermatology*, `https://www.americantelemed.org/wp-content/themes/ata-custom/download.php?id=1559`, Accessed: 2020-12-16.

[73] J. J. Lee and J. C. English, "Teledermatology: A review and update," *American Journal of Clinical Dermatology*, vol. 19, no. 2, pp. 253–260, 2018.

[74] Diário da República Eletrónico, *Despacho n.º 6280/2018*, `https://dre.pt/web/guest/home/-/dre/115600144/details/2/maximized?serie=II&parte_filter=31&day=2018-06-28&date=2018-06-01&dreId=115600115`, Accessed: 2020-12-16.

[75] American Academy of Dermatology Association, *Position Statement on Teledermatology*, `https://server.aad.org/Forms/Policies/Uploads/PS/PS-Teledermatology.pdf?`, Accessed: 2021-01-14.

[76] L. M. Abbott, R. Miller, M. Janda, H. Bennett, M. Taylor, C. Arnold, S. Shumack, H. P. Soyer, and L. J. Caffery, "Practice guidelines for teledermatology in australia," *Australasian Journal of Dermatology*, 2020.

[77] S. Khan and S.-P. Yong, "A comparison of deep learning and hand crafted features in medical image modality classification," in *2016 3rd International Conference on Computer and Information Sciences (ICCOINS)*, IEEE, 2016, pp. 633–638.

[78] S. Singh, K. Ho-Shon, S. Karimi, and L. Hamey, "Modality classification and concept detection in medical images using deep transfer learning," in *2018 International Conference on Image and Vision Computing New Zealand (IVCNZ)*, IEEE, 2018, pp. 1–9.

[79] M. Hassan, S. Ali, H. Alquhayz, and K. Safdar, "Developing intelligent medical image modality classification system using deep transfer learning and lda," *Scientific reports*, vol. 10, no. 1, pp. 1–14, 2020.

[80] ImageCLEF, *Medical Image Classification and Retrieval 2012*, `https://www.imageclef.org/2012/medical`, Accessed: 2020-12-16.

[81] G. Csurka, S. Clinchant, and G. Jacquet, "Medical image modality classification and retrieval," in *2011 9th International Workshop on Content-Based Multimedia Indexing (CBMI)*, IEEE, 2011, pp. 193–198.

[82] I. Dimitrovski, D. Kocev, I. Kitanovski, S. Loskovska, and S. Džeroski, "Improved medical image modality classification using a combination of visual and textual features," *Computerized Medical Imaging and Graphics*, vol. 39, pp. 14–26, 2015.

[83] J. Kalpathy-Cramer, W. Hersh, *et al.*, "Automatic image modality based classification and annotation to improve medical image retrieval," in *Medinfo 2007: Proceedings of the 12th World Congress on Health (Medical) Informatics; Building Sustainable Health Systems*, IOS Press, 2007, p. 1334.

[84] B. P. Hibler, Q. Qi, and A. M. Rossi, "Current state of imaging in dermatology," *Semin Cutan Med Surg*, vol. 35, no. 1, pp. 2–8, 2016.

[85] S. L. Schneider, I. Kohli, I. H. Hamzavi, M. L. Council, A. M. Rossi, and D. M. Ozog, "Emerging imaging technologies in dermatology: Part I: Basic principles," *Journal of the American Academy of Dermatology*, vol. 80, no. 4, pp. 1114–1120, 2019.

[86] S. L. Schneider, I. Kohli, I. H. Hamzavi, M. L. Council, A. M. Rossi, and D. M. Ozog, "Emerging imaging technologies in dermatology: Part II: Applications and limitations," *Journal of the American Academy of Dermatology*, vol. 80, no. 4, pp. 1121–1131, 2019.

[87] K. J. Lee and H. P. Soyer, "Future developments in teledermoscopy and total body photography," *International Journal of Dermatology and Venereology*, vol. 2, no. 1, pp. 15–18, 2019.

[88] H. K. Flaten, C. St Claire, E. Schlager, C. A. Dunnick, and R. P. Dellavalle, "Growth of mobile applications in dermatology-2017 update," *Dermatology online journal*, vol. 24, no. 2, 2018.

[89] S. Q. Wang, A. W. Kopf, K. Koenig, D. Polsky, K. Nudel, and R. S. Bart, "Detection of melanomas in patients followed up with total cutaneous examinations, total cutaneous photography, and dermoscopy," *Journal of the American Academy of Dermatology*, vol. 50, no. 1, pp. 15–20, 2004.

[90] A. Ji-Xu, J. Dinnes, and R. Matin, "Total body photography for the diagnosis of cutaneous melanoma in adults: A systematic review and meta-analysis," *British Journal of Dermatology*, 2020.

[91] R. B. Oliveira, N. Marranghello, A. S. Pereira, and J. M. R. Tavares, "A computational approach for detecting pigmented skin lesions in macroscopic images," *Expert Systems with Applications*, vol. 61, pp. 53–63, 2016.

[92] I. Zalaudek, G. Argenziano, A. Di Stefani, G. Ferrara, A. A. Marghoob, R. Hofmann-Wellenhof, H. P. Soyer, R. Braun, and H. Kerl, "Dermoscopy in general dermatology," *Dermatology*, vol. 212, no. 1, pp. 7–18, 2006.

[93] C. Benvenuto-Andrade, S. W. Dusza, A. L. C. Agero, A. Scope, M. Rajadhyaksha, A. C. Halpern, and A. A. Marghoob, "Differences between polarized light dermoscopy and immersion contact dermoscopy for the evaluation of skin lesions," *Archives of dermatology*, vol. 143, no. 3, pp. 329–338, 2007.

[94] S. J. Coates, J. Kvedar, and R. D. Granstein, "Teledermatology: From historical perspective to emerging techniques of the modern era: Part ii: Emerging technologies in teledermatology, limitations and future directions," *Journal of the American Academy of Dermatology*, vol. 72, no. 4, pp. 577–586, 2015.

[95] X.-D. Zhang, "Machine learning," in *A Matrix Algebra Approach to Artificial Intelligence*, Springer, 2020, pp. 223–440.

[96] G. Rebala, A. Ravi, and S. Churiwala, "Machine learning definition and basics," in *An Introduction to Machine Learning*, Springer, 2019, pp. 1–17.

[97] D. Sarkar, R. Bali, and T. Sharma, "Machine learning basics," in *Practical Machine Learning with Python*, Springer, 2018, pp. 3–65.

[98] M. Kang and N. J. Jameson, "Machine learning: Fundamentals," *Prognostics and Health Management of Electronics: Fundamentals, Machine Learning, and the Internet of Things*, pp. 85–109, 2018.

[99] T. Qin, "Machine learning basics," in *Dual Learning*, Springer, 2020, pp. 11–23.

[100] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of machine learning*. MIT press, 2018.

[101] F. Hu and Q. Hao, *Intelligent sensor networks: the integration of sensor networks, signal processing and machine learning*. Taylor & Francis, 2012.

[102] I. G. Maglogiannis, *Emerging artificial intelligence applications in computer engineering: real word ai systems with applications in ehealth, hci, information retrieval and pervasive technologies*. Ios Press, 2007, vol. 160.

[103] D. G. Kleinbaum, K. Dietz, M. Gail, M. Klein, and M. Klein, *Logistic regression*. Springer, 2002.

[104] S. Dreiseitl and L. Ohno-Machado, "Logistic regression and artificial neural network classification models: A methodology review," *Journal of biomedical informatics*, vol. 35, no. 5-6, pp. 352–359, 2002.

[105] S. B. Kotsiantis, I. D. Zaharakis, and P. E. Pintelas, "Machine learning: A review of classification and combining techniques," *Artificial Intelligence Review*, vol. 26, no. 3, pp. 159–190, 2006.

[106] F. F. Chamasemani and Y. P. Singh, "Multi-class support vector machine (svm) classifiers–an application in hypothyroid detection and classification," in *2011 Sixth International Conference on Bio-Inspired Computing: Theories and Applications*, IEEE, 2011, pp. 351–356.

[107] R. Gandhi, *Support vector machine — introduction to machine learning algorithms*, `https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47`, Accessed: 2021-01-31.

[108] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*, 2. 2016, vol. 1.

[109] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[110] S.-C. Wang, "Artificial neural network," in *Interdisciplinary computing in java programming*, Springer, 2003, pp. 81–100.

[111] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, pp. 1097–1105, 2012.

[112] Z. Qin, F. Yu, C. Liu, and X. Chen, "How convolutional neural network see the world-a survey of convolutional neural network visualization methods," *arXiv preprint arXiv:1804.11191*, 2018.

[113] S. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," in *2017 International Conference on Engineering and Technology (ICET)*, Ieee, 2017, pp. 1–6.

[114] K. O'Shea and R. Nash, "An introduction to convolutional neural networks," *arXiv preprint arXiv:1511.08458*, 2015.

[115] T. Liu, S. Fang, Y. Zhao, P. Wang, and J. Zhang, "Implementation of training convolutional neural networks," *arXiv preprint arXiv:1506.01195*, 2015.

[116] V. H. Phung, E. J. Rhee, *et al.*, "A high-accuracy model average ensemble of convolutional neural networks for classification of cloud image patches on small datasets," *Applied Sciences*, vol. 9, no. 21, p. 4500, 2019.

[117] P. Kim, "Convolutional neural network," in *MATLAB deep learning*, Springer, 2017, pp. 121–147.

[118] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: An overview and application in radiology," *Insights into imaging*, vol. 9, no. 4, pp. 611–629, 2018.

[119] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[120] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[121] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[122] 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.

[123] 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.

[124] N. Qian, "On the momentum term in gradient descent learning algorithms," *Neural networks*, vol. 12, no. 1, pp. 145–151, 1999.

[125] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization.," *Journal of machine learning research*, vol. 12, no. 7, 2011.

[126] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, *et al.*, "Imagenet large scale visual recognition challenge," *International journal of computer vision*, vol. 115, no. 3, pp. 211–252, 2015.

[127] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 31, 2017.

[128] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[129] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.

[130] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International Conference on Machine Learning*, PMLR, 2019, pp. 6105–6114.

[131] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7132–7141.

[132] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv: 1609.04747*, 2016.

[133] L. Torrey and J. Shavlik, "Transfer learning," in *Handbook of research on machine learning applications and trends: algorithms, methods, and techniques*, IGI global, 2010, pp. 242–264.

[134] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *Journal of Big data*, vol. 3, no. 1, pp. 1–40, 2016.

[135] X. Wu, D. Sahoo, and S. C. Hoi, "Recent advances in deep learning for object detection," *Neurocomputing*, vol. 396, pp. 39–64, 2020.

[136] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, and M. Pietikäinen, "Deep learning for generic object detection: A survey," *International journal of computer vision*, vol. 128, no. 2, pp. 261–318, 2020.

[137] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.

[138] L. Jiao, F. Zhang, F. Liu, S. Yang, L. Li, Z. Feng, and R. Qu, "A survey of deep learning-based object detection," *IEEE Access*, vol. 7, pp. 128 837–128 868, 2019.

[139] Z. Zou, Z. Shi, Y. Guo, and J. Ye, "Object detection in 20 years: A survey," *arXiv preprint arXiv:1905.05055*, 2019.

[140] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[141] J. R. Uijlings, K. E. Van De Sande, T. Gevers, and A. W. Smeulders, "Selective search for object recognition," *International journal of computer vision*, vol. 104, no. 2, pp. 154–171, 2013.

[142] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 9, pp. 1904–1916, 2015.

[143] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.

[144] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *arXiv preprint arXiv:1506.01497*, 2015.

[145] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[146] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[147] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *European conference on computer vision*, Springer, 2016, pp. 21–37.

[148] M. Tan, R. Pang, and Q. V. Le, "Efficientdet: Scalable and efficient object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 10 781–10 790.

[149] J. Redmon and A. Farhadi, "YOLO9000: better, faster, stronger," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7263–7271.

[150] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[151] S. S. Sarwar, A. Ankit, and K. Roy, "Incremental learning in deep convolutional neural networks using partial network sharing," *IEEE Access*, vol. 8, pp. 4615–4628, 2019.

[152] J. Hu, C. Yan, X. Liu, Z. Li, C. Ren, J. Zhang, D. Peng, and Y. Yang, "An integrated classification model for incremental learning," *Multimedia Tools and Applications*, pp. 1–16, 2020.

[153] M. De Lange, R. Aljundi, M. Masana, S. Parisot, X. Jia, A. Leonardis, G. Slabaugh, and T. Tuytelaars, "A continual learning survey: Defying forgetting in classification tasks," *arXiv preprint arXiv:1909.08383*, 2019.

[154] Z. Mai, R. Li, J. Jeong, D. Quispe, H. Kim, and S. Sanner, "Online continual learning in image classification: An empirical survey," 2021.

[155] T. Lesort, V. Lomonaco, A. Stoian, D. Maltoni, D. Filliat, and N. Dıaz-Rodrıguez, "Continual learning for robotics: Definition, framework, learning strategies, opportunities and challenges," *Information fusion*, vol. 58, pp. 52–68, 2020.

[156] M. McCloskey and N. J. Cohen, "Catastrophic interference in connectionist networks: The sequential learning problem," in *Psychology of learning and motivation*, vol. 24, Elsevier, 1989, pp. 109–165.

[157] A. Bouchachia, B. Gabrys, and Z. Sahel, "Overview of some incremental learning algorithms," in *2007 IEEE International Fuzzy Systems Conference*, IEEE, 2007, pp. 1–6.

[158] A. Gaurav, S. Vernekar, J. Lee, V. Abdelzad, K. Czarnecki, and S. Sedwards, "Simple continual learning strategies for safer classifers.," in *SafeAI@ AAAI*, 2020, pp. 96–104.

[159] M. Mermillod, A. Bugaiska, and P. Bonin, "The stability-plasticity dilemma: Investigating the continuum from catastrophic forgetting to age-limited learning effects," *Frontiers in psychology*, vol. 4, p. 504, 2013.

[160] A. Gepperth and B. Hammer, "Incremental learning algorithms and applications," in *European symposium on artificial neural networks (ESANN)*, 2016.

[161] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, and S. Wermter, "Continual lifelong learning with neural networks: A review," *Neural Networks*, vol. 113, pp. 54–71, 2019.

[162] F. M. Castro, M. J. Marın-Jiménez, N. Guil, C. Schmid, and K. Alahari, "End-to-end incremental learning," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 233–248.

[163] J. He, R. Mao, Z. Shao, and F. Zhu, "Incremental learning in online scenario," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 13 926–13 935.

[164] A. Awasthi and S. Sarawagi, "Continual learning with neural networks: A review," in *Proceedings of the ACM India Joint International Conference on Data Science and Management of Data*, 2019, pp. 362–365.

[165] P. Joshi and P. Kulkarni, "Incremental learning: Areas and methods-a survey," *International Journal of Data Mining & Knowledge Management Process*, vol. 2, no. 5, p. 43, 2012.

[166] K. Lee, K. Lee, J. Shin, and H. Lee, "Incremental learning with unlabeled data in the wild.," in *CVPR Workshops*, 2019, pp. 29–32.

[167] R. Kemker, M. McClure, A. Abitino, T. Hayes, and C. Kanan, "Measuring catastrophic forgetting in neural networks," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, 2018.

[168] R. Polikar, L. Upda, S. S. Upda, and V. Honavar, "Learn++: An incremental learning algorithm for supervised neural networks," *IEEE transactions on systems, man, and cybernetics, part C (applications and reviews)*, vol. 31, no. 4, pp. 497–508, 2001.

[169] V. Lomonaco and D. Maltoni, "Core50: A new dataset and benchmark for continuous object recognition," in *Conference on Robot Learning*, PMLR, 2017, pp. 17–26.

[170] A. Douillard, *Learning deep neural networks incrementally*, `https://medium.com/heuritech/learning-deep-neural-networks-incrementally-3e005e4fb4bc`, Accessed: 2021-01-11.

[171] F. Zenke, B. Poole, and S. Ganguli, "Continual learning through synaptic intelligence," in *International Conference on Machine Learning*, PMLR, 2017, pp. 3987–3995.

[172] J. Hofmanninger, M. Perkonigg, J. A. Brink, O. Pianykh, C. Herold, and G. Langs, "Dynamic memory to alleviate catastrophic forgetting in continuous learning settings," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Springer, 2020, pp. 359–368.

[173] G. M. Van de Ven and A. S. Tolias, "Three scenarios for continual learning," *arXiv preprint arXiv:1904.07734*, 2019.

[174] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.

[175] T. Fawcett, "An introduction to roc analysis," *Pattern recognition letters*, vol. 27, no. 8, pp. 861–874, 2006.

[176] J. Fan, S. Upadhye, and A. Worster, "Understanding receiver operating characteristic (roc) curves," *Canadian Journal of Emergency Medicine*, vol. 8, no. 1, pp. 19–20, 2006.

[177] D. Lopez-Paz and M. Ranzato, "Gradient episodic memory for continual learning," *arXiv preprint arXiv:1706.08840*, 2017.

[178] N. Dıaz-Rodrıguez, V. Lomonaco, D. Filliat, and D. Maltoni, "Don't forget, there is more than forgetting: New metrics for continual learning," *arXiv preprint arXiv:1810.13166*, 2018.

[179] A. Chaudhry, P. K. Dokania, T. Ajanthan, and P. H. Torr, "Riemannian walk for incremental learning: Understanding forgetting and intransigence," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 532–547.

[180] W. Song, D. Zhang, and J. Luo, "BUAA AUDR at ImageCLEF 2012 Medical Retrieval Task.," in *CLEF (Online Working Notes/Labs/Workshop)*, 2012.

[181] M. Y. Khachane and R. Ramteke, "Modality based medical image classification," in *Emerging research in computing, information, communication and applications*, Springer, 2016, pp. 597–606.

[182] J. Arias, J. Martinez-Gomez, J. A. Gamez, A. G. S. de Herrera, and H. Müller, "Medical image modality classification using discrete Bayesian networks," *Computer vision and image understanding*, vol. 151, pp. 61–71, 2016.

[183] J. Yang, Y.-G. Jiang, A. G. Hauptmann, and C.-W. Ngo, "Evaluating bag-of-visual-words representations in scene classification," in *Proceedings of the international workshop on Workshop on multimedia information retrieval*, 2007, pp. 197–206.

[184] L. Cao, Y.-C. Chang, N. Codella, M. Merler, Q.-B. Nguyen, and J. R. Smith, "Multimedia Analytics: Modality Classification and Case-Based Retrieval tasks of ImageCLEF2012,"

[185] L. Valavanis, S. Stathopoulos, and T. Kalamboukis, "IPL at CLEF 2016 medical task.," in *CLEF (Working Notes)*, 2016, pp. 413–420.

[186] D. Markonis, I. Eggel, A. G. S. de Herrera, and H. Müller, "The medGIFT Group in ImageCLEFmed 2011.," in *CLEF (Notebook Papers/Labs/Workshop)*, 2011.

[187] I. Kitanovski, I. Dimitrovski, and S. Loskovska, "FCSE at Medical Tasks of ImageCLEF 2013.," in *CLEF (Working Notes)*, 2013.

[188] O. Pelka and C. M. Friedrich, "FHDO Biomedical Computer Science Group at Medical Classification Task of ImageCLEF 2015.," *CLEF (Working Notes)*, vol. 1391, 2015.

[189] O. Pelka and C. M. Friedrich, "Modality prediction of biomedical literature images using multimodal feature representation.," *GMS Medizinische Informatik, Biometrie und Epidemiologie*, vol. 12, no. 2, 2016.

[190] H. Wu and L. He, "Combining visual and textual features for medical image modality classification with p- norm multiple kernel learning," *Neurocomputing*, vol. 147, pp. 387–394, 2015.

[191] V. Gál, I. Solt, T. Gedeon, and M. Nachtegael, "Multi-disciplinary modality classification for medical images," *magnetic resonance imaging*, vol. 17, pp. 1–7, 2011.

[192] A. Kumar, J. Kim, D. Lyndon, M. Fulham, and D. Feng, "An ensemble of fine-tuned convolutional neural networks for medical image classification," *IEEE journal of biomedical and health informatics*, vol. 21, no. 1, pp. 31–40, 2016.

[193] A. Kumar, D. Lyndon, J. Kim, and D. Feng, "Subfigure and Multi-Label Classification using a Fine-Tuned Convolutional Neural Network.," in *CLEF (Working Notes)*, 2016, pp. 318–321.

[194] D. Semedo and J. Magalhães, "NovaSearch at ImageCLEFmed 2016 Subfigure Classification Task.," in *CLEF (Working Notes)*, 2016, pp. 386–398.

[195] L. Wan, M. Zeiler, S. Zhang, Y. Le Cun, and R. Fergus, "Regularization of neural networks using dropconnect," in *International conference on machine learning*, 2013, pp. 1058–1066.

[196] Y. Yu, H. Lin, J. Meng, X. Wei, H. Guo, and Z. Zhao, "Deep transfer learning for modality classification of medical images," *Information*, vol. 8, no. 3, p. 91, 2017.

[197] J. Zhang, Y. Xia, Q. Wu, and Y. Xie, "Classification of medical images and illustrations in the biomedical literature using synergic deep learning," *arXiv preprint arXiv:1706.09092*, 2017.

[198] V. Pomponiu, H. Nejati, and N.-M. Cheung, "Deepmole: Deep neural networks for skin mole lesion classification," in *2016 IEEE International Conference on Image Processing (ICIP)*, IEEE, 2016, pp. 2623–2627.

[199] N. Codella, J. Cai, M. Abedini, R. Garnavi, A. Halpern, and J. R. Smith, "Deep learning, sparse coding, and svm for melanoma recognition in dermoscopy images," in *International workshop on machine learning in medical imaging*, Springer, 2015, pp. 118–126.

[200] J. Kawahara, A. BenTaieb, and G. Hamarneh, "Deep features to classify skin lesions," in *2016 IEEE 13th international symposium on biomedical imaging (ISBI)*, IEEE, 2016, pp. 1397–1400.

[201] 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.

[202] S. S. Han, M. S. Kim, W. Lim, G. H. Park, I. Park, and S. E. Chang, "Classification of the clinical images for benign and malignant cutaneous tumors using a deep learning algorithm," *Journal of Investigative Dermatology*, vol. 138, no. 7, pp. 1529–1538, 2018.

[203] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 618–626.

[204] M. A. Marchetti, N. C. Codella, S. W. Dusza, D. A. Gutman, B. Helba, A. Kalloo, N. Mishra, C. Carrera, M. E. Celebi, J. L. DeFazio, *et al.*, "Results of the 2016 international skin imaging collaboration international symposium on biomedical imaging challenge: Comparison of the accuracy of computer algorithms to dermatologists for the diagnosis of melanoma from dermoscopic images," *Journal of the American Academy of Dermatology*, vol. 78, no. 2, pp. 270–277, 2018.

[205] L. Bi, J. Kim, E. Ahn, and D. Feng, "Automatic skin lesion analysis using large-scale dermoscopy images and deep residual networks," *arXiv preprint arXiv:1703.04197*, 2017.

[206] J. Kawahara and G. Hamarneh, "Multi-resolution-tract cnn with hybrid pretrained and skin-lesion trained layers," in *International workshop on machine learning in medical imaging*, Springer, 2016, pp. 164–171.

[207] X. Sun, J. Yang, M. Sun, and K. Wang, "A benchmark for automatic visual classification of clinical skin disease images," in *European Conference on Computer Vision*, Springer, 2016, pp. 206–222.

[208] A. R. Lopez, X. Giro-i-Nieto, J. Burdick, and O. Marques, "Skin lesion classification from dermoscopic images using deep learning techniques," in *2017 13th IASTED international conference on biomedical engineering (BioMed)*, IEEE, 2017, pp. 49–54.

[209] E. Nasr-Esfahani, S. Samavi, N. Karimi, S. M. R. Soroushmehr, M. H. Jafari, K. Ward, and K. Najarian, "Melanoma detection by analysis of clinical images using convolutional neural network," in *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, IEEE, 2016, pp. 1373–1376.

[210] S. Vesal, S. M. Patil, N. Ravikumar, and A. K. Maier, "A multi-task framework for skin lesion detection and segmentation," in *OR 2.0 Context-Aware Operating Theaters, Computer Assisted Robotic Endoscopy, Clinical Image-Based Procedures, and Skin Image Analysis*, Springer, 2018, pp. 285–293.

[211] S. Vesal, N. Ravikumar, and A. Maier, "Skinnet: A deep learning framework for skin lesion segmentation," in *2018 IEEE Nuclear Science Symposium and Medical Imaging Conference Proceedings (NSS/MIC)*, IEEE, 2018, pp. 1–3.

[212] C. Qian, T. Liu, H. Jiang, Z. Wang, P. Wang, M. Guan, and B. Sun, "A detection and segmentation architecture for skin lesion segmentation on dermoscopy images," *arXiv preprint arXiv:1809.03917*, 2018.

[213] H. M. Ünver and E. Ayan, "Skin lesion segmentation in dermoscopic images with combination of yolo and grabcut algorithm," *Diagnostics*, vol. 9, no. 3, p. 72, 2019.

[214] M. A. Khan, T. Akram, Y.-D. Zhang, and M. Sharif, "Attributes based skin lesion detection and recognition: A mask rcnn and transfer learning-based deep learning framework," *Pattern Recognition Letters*, 2021.

[215] D. Bisla, A. Choromanska, R. S. Berman, J. A. Stein, and D. Polsky, "Towards automated melanoma detection with deep learning: Data purification and augmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2019, pp. 0–0.

[216] P. Sabouri and H. GholamHosseini, "Lesion border detection using deep learning," in *2016 IEEE Congress on Evolutionary Computation (CEC)*, IEEE, 2016, pp. 1416–1421.

[217] A. A. Rusu, N. C. Rabinowitz, G. Desjardins, H. Soyer, J. Kirkpatrick, K. Kavukcuoglu, R. Pascanu, and R. Hadsell, "Progressive neural networks," *arXiv preprint arXiv:1606.04671*, 2016.

[218] R. Aljundi, P. Chakravarty, and T. Tuytelaars, "Expert gate: Lifelong learning with a network of experts," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3366–3375.

[219] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, *et al.*, "Overcoming catastrophic forgetting in neural networks," *Proceedings of the national academy of sciences*, vol. 114, no. 13, pp. 3521–3526, 2017.

[220] Z. Li and D. Hoiem, "Learning without forgetting," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 12, pp. 2935–2947, 2017.

[221] S.-A. Rebuffi, A. Kolesnikov, G. Sperl, and C. H. Lampert, "Icarl: Incremental classifier and representation learning," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017, pp. 2001–2010.

[222] R. Kemker and C. Kanan, "Fearnet: Brain-inspired model for incremental learning," *arXiv preprint arXiv:1711.10563*, 2017.

[223] A. Chaudhry, M. Ranzato, M. Rohrbach, and M. Elhoseiny, "Efficient lifelong learning with a-gem," *arXiv preprint arXiv:1812.00420*, 2018.

[224] A. Chaudhry, M. Rohrbach, M. Elhoseiny, T. Ajanthan, P. K. Dokania, P. H. Torr, and M. Ranzato, "On tiny episodic memories in continual learning," *arXiv preprint arXiv:1902.10486*, 2019.

[225] Q. Meng and S. Shin'ichi, "Adinet: Attribute driven incremental network for retinal image classification," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 4033–4042.

[226] H. Ravishankar, R. Venkataramani, S. Anamandra, P. Sudhakar, and P. Annangi, "Feature transformers: Privacy preserving lifelong learners for medical imaging," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Springer, 2019, pp. 347–355.

[227] K. van Garderen, S. van der Voort, F. Incekara, M. Smits, and S. Klein, "Towards continuous learning for glioma segmentation with elastic weight consolidation," *arXiv preprint arXiv:1909.11479*, 2019.

[228] N. Karani, K. Chaitanya, C. Baumgartner, and E. Konukoglu, "A lifelong learning approach to brain mr segmentation across scanners and protocols," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Springer, 2018, pp. 476–484.

[229] A. Tang, R. Tam, A. Cadrin-Chênevert, W. Guest, J. Chong, J. Barfett, L. Chepelev, R. Cairns, J. R. Mitchell, M. D. Cicero, *et al.*, "Canadian association of radiologists white paper on artificial intelligence in radiology," *Canadian Association of Radiologists Journal*, vol. 69, no. 2, pp. 120–135, 2018.

[230] COCO, *COCO - Common Objects in Context*, `https://cocodataset.org/#home`, Accessed: 2021-04-04.

[231] V. Lomonaco, L. Pellegrini, A. Cossu, A. Carta, G. Graffieti, T. L. Hayes, M. D. Lange, M. Masana, J. Pomponi, G. van de Ven, M. Mundt, Q. She, K. Cooper, J. Forest, E. Belouadah, S. Calderara, G. I. Parisi, F. Cuzzolin, A. Tolias, S. Scardapane, L. Antiga, S. Amhad, A. Popescu, C. Kanan, J. van de Weijer, T. Tuytelaars, D. Bacciu, and D. Maltoni, "Avalanche: An end-to-end library for continual learning," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, ser. 2nd Continual Learning in Computer Vision Workshop, 2021.

[232] E. Verwimp, M. De Lange, and T. Tuytelaars, "Rehearsal revealed: The limits and merits of revisiting samples in continual learning," *arXiv preprint arXiv:2104.07446*, 2021.