Automatic Prediction of BRAF Mutation in Melanoma Using Deep Learning

Simão Campos Gonçalves
simao.goncalves@tecnico.ulisboa.pt
Instituto Superior Técnico, Lisboa, Portugal

***Abstract*—Melanoma is the deadliest form of skin cancer. The treatment of metastatic melanoma patients depends on the mutational state of the BRAF gene. Thus, it is crucial to timely assess this gene's status to select an adequate treatment. Nowadays, to infer the BRAF status, a biopsy is performed on the lesion area, and after that, PCR analysis is executed on the extracted DNA. This process is efficient; however, it is slow and depends on the experience of specialized personnel. It is essential to complement the existing diagnostic techniques. Previous works have shown that dermoscopic images of melanomas convey relevant information about the BRAF mutational status. The objective of this thesis is to explore faster, more automated, and less human-dependent ways of predicting BRAF status for melanoma patients. Regarding the BRAF data, only a few labeled *ex vivo* dermoscopic images are available for this work. Therefore, three convolutional neural networks are pre-trained on a related task (benign/melanoma classification task) using a larger *in vivo* dataset. Some versions of the related task pre-training use *ex vivo* data to perform domain adaptation, attempting to mitigate the shift between the *in vivo* and *ex vivo* data. The pre-trained architectures are used to extract features from the BRAF dataset, and classification algorithms are employed to predict the mutational status, most inspired by few-shot learning. The results obtained in this work overcome the current state-of-the-art results, proving that the proposed deep learning approaches are a promising venue of research for BRAF status prediction.**

***Index Terms*—Deep Learning, Few-Shot Learning, Domain Adaptation, Melanoma, BRAF, Dermoscopy**

I. Introduction

A significant part of deaths related to skin cancer is associated with melanoma. Usually, to treat a melanoma patient, surgical excision is proposed. However, when the tumor is metastatic, other treatment options must be explored. Chemotherapy, immunotherapy, and targeted therapies are possible options for these cases [1]. Target therapies are one type of treatment that has emerged in the last years, as well as immunotherapy. These treatments allow acting in specific genetic mutations within the cells, promoting personalized therapies. Nowadays, the type of treatment indicated to a patient who suffers from metastatic melanoma varies according to BRAF status in the cancerous cells [2]. The BRAF gene can be either classified as Non-Mutated BRAF (BRAF-) or as Mutated BRAF (BRAF+). BRAF is a gene that controls cell growth. Therefore, when mutated, it leads to a more aggressive expansion of the tumor [2].

Currently, to evaluate the mutational status of BRAF, the most common approach is followed by an excision biopsy. DNA extracted from cells of the skin lesion is evaluated through PCR analysis which is a slow procedure and requires the work of specialized personnel. The analysis depends on the pathologist's experience, as the pathologist must select the most relevant part of the lesion to be submitted to the PCR analysis. Different parts of the lesion might lead to different outcomes. Therefore, more than one PCR might be needed to confirm the prediction.

The search for alternative ways to check on the mutational status of the BRAF gene is justified because it is crucial to find faster, more automatized, and less human-dependent ways of inferring this gene's status.

Some studies show that different medical images may convey relevant information regarding cancer characteristics. For instance, Brinker *et al.* [3] explored a Deep Learning (DL) approach to address Sentinel Lymph Node (SLN) status for melanoma patients. The authors proposed a pipeline that combined various features (clinical features, cell features, and image features). The idea was to use a Convolutional Neural Network (CNN) to extract melanoma features from prior melanoma tiles. The used tiles were obtained by tiling Hematoxylin and Eosin (HE) stained Whole Slide Images (WSIs) that were annotated by a bioinformatician. The image features extracted by the CNN could be combined afterward with the clinical and cell features, which in turn, were processed by a Multilayer Perceptron (MLP). The work of Armengot-Carbó *et al.* [4] used dermoscopic images to predict BRAF status in melanoma patients. Dermoscopy is a skin surface microscopy technique that allows obtaining high-resolution images of skin lesions, in a non-invasive way. The main objective of the work in [4] was to relate BRAF status with dermoscopic features, which is still a poorly explored subject. In the conducted study, dermoscopic images from cutaneous melanomas were considered, as well as clinical and histopathological data. From this work, it was concluded that the dermoscopic features more frequent on BRAF+ melanomas were streaks, exophytic papillary structures, and BWVs. Thus, providing evidence that some dermoscopic features relate to BRAF mutational status. Furthermore, BRAF+ was more incident in younger patients, and the BWV structure was the one feature that seemed to relate more with the mutation as it was the one that kept bigger statistical significance after performing multivariate analysis. The authors developed a classification tree that considered only two variables: patient age and the presence/absence of BWVs in the dermoscopic images, attaining an accuracy of 73.1%.

This thesis aims to automatize the inspection of dermo-

scopic images and to reduce the subjectivity associated with human analysis. To this end, the use of DL methods to extract relevant information (features) for BRAF status prediction from dermoscopic images is proposed. These features can then be processed by other Machine Learning (ML) algorithms to predict the presence/absence of mutation. DL methods require a high volume of data to achieve reliability. Unfortunately, the only available dermoscopic dataset with BRAF information is small. However, Few-Shot Learning (FSL) strategies intend to overcome this problem, making it a topic worth of exploring. Besides, for this research, *in vivo* as well as *ex vivo* dermoscopic images are present so it will be interesting to explore if a model can perform well in the different domains. Some data processing techniques address issues involving the different nature of datasets. One well-known strategy to address this matter is Domain Adaptation (DA).

The main objective of this work is to both accelerate and automatize BRAF status prediction using DL based approaches, reducing the dependence on human expertise. Fulfilling this objective will complement the existing ways to assess BRAF status and will speed up the process of selecting the most appropriate treatment for each patient which, consequently, increases the survival expectation. Other scientific objectives are stated, namely to study if techniques based on FSL will help overcoming the lack of data and if DA methods will be helpful in a context were more than one domain is involved.

II. Theoretical Background

A. Few-Shot Learning

ML, in particular DL, has shown to be effective when a large amount of data is available for training. However, when the datasets are small, ML methods struggle to generalize. Modern learning approaches, such as FSL, try to overcome this problem. This learning mechanism uses prior knowledge to obtain good performance on a target task given only a few labeled data during the training stage.

As previously referred, in this work, there will be very few labels regarding the BRAF status in the dermoscopic images of the melanomas so supervised FSL can be useful to obtain an adequate classifier.

FSL differs from traditional supervised learning. In the traditional approach, a model is trained on a large dataset. Then in the testing stage, the model is presented with unseen examples and classifies those examples in one of the classes seen during the training stage. In FSL, the objective is different; instead of having as the main goal the classification of unseen data in known classes (base classes), the objective can be to learn a model capable of classifying data in novel classes, i.e., classes not present during the training stage, given only a few labeled examples. Another objective of FSL can also be to learn a model that can perform classification tasks similar to the ones of the training stage but with fewer data. In this work, first, a general description of meta-learning is supplemented, and after that, metric-based methods for FSL are reviewed.

1) ***Meta-learning****:* Learning mechanism present in a gross part of FSL algorithms. It comprehends two stages, a meta-training stage and a meta-testing stage. In the meta-training stage, training can be divided into various episodes, for instance, several classification events where a meta-learner trained on a few examples from various classes tries to classify an unseen sample in one of those classes [5]. The objective is for the meta-learner to improve its capabilities of classifying unseen samples in base classes by acquiring knowledge from different classification tasks [6]. In the meta-testing stage, other classification tasks are presented to the meta-learner. Here, a few labeled images from novel classes are shown to the model to conclude if it can perform well in these situations. Using a similar notation to the one present in [5], each meta-training episode i consists of a training set D^i_{train} (also called support set) and a test set D^i_{test} (also called query set) where the classifier performance is measured. A meta-learner learns from various episodic tasks T_i. These episodic tasks are often N-way-K-shot classification tasks, i.e., the support sets consist of N classes, each class with K examples [6]. Figure 1 exemplifies a 3-way-2-shot-classification problem.

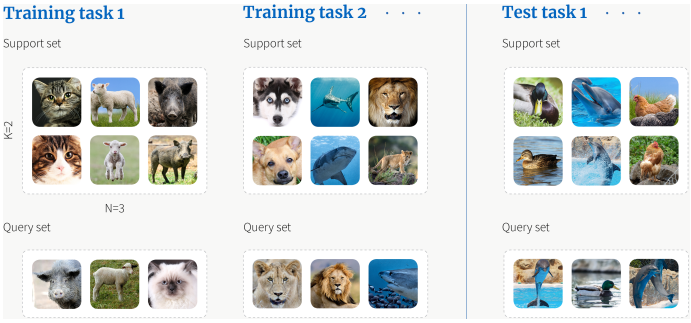

Fig. 1. Meta-learning framework: 3-way-2-shot classification episodes [6].

2) ***Metric-based FSL algorithms****:* Comprises FSL algorithms that use distance metrics to compare the similarity of two images. The MatchingNet [7] (figure 2) and the ProtoNet [8] (figure 3) use distance metric-based classifiers that are put together with the concept of meta-learning, producing FSL algorithms.

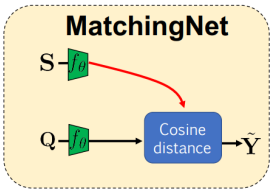

Fig. 2. MatchingNet [9].

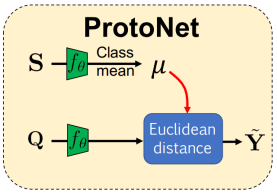

Fig. 3. ProtoNet [9].

In both approaches, a CNN works as a feature extractor (f_θ) on the images from the support set (S) and the query set (Q).

The extractor is parametrized by the network's parameters θ. As previously explained, in meta-learning, the available data is sampled and divided into several classification tasks. On each task, there will be a few examples of some base classes on the support set. In the case of a MatchingNet, for each task, the features of the images on the support set are compared to those of the query set by the cosine similarity. On top of that, the average cosine similarity is calculated for each class. In the case of a ProtoNet, the idea is to use the feature vectors extracted from the support set to compute class prototypes and then to measure the euclidean distance from each prototype to the features extracted from each query set image.

B. Domain Adaptation

In the context of this work, DA techniques will be used because one of the challenges is to build a model which can perform well on datasets that come from different domains (*in vivo* and *ex vivo*). When it comes to medical imaging, the majority of ML methods consider that the data distribution in the training and test sets is the same, which is deceptive most of the time. This assumption results in worse performance for a model. The increase of test error as the distribution difference between the training and test sets accentuates is called the domain shift problem. The domain shift problem is associated with models which, for instance, analyze the same type of medical image but from different data centers (multi-site data). In this situation, the model is affected because the datasets are obtained using different scan technologies and/or methods. DA techniques are useful because they allow to minimize the distribution difference between different but related domains [10].

DA can be seen as a particular case of transfer learning. Transfer learning is, according to [11], the concept in which a model is trained in a specific domain (source domain) or task and evaluated in a different domain (target domain) or task. The domain is defined as a feature space together with a marginal probability distribution and a task as a group of labels together with a predictive function learned in the training process [12]. In the DA case, the task is the same in both the source and the target domain, but the marginal distribution of features within the domains differs [13]. In [10], DA procedures are categorized as either shallow or deep and additionally as supervised, semi-supervised and unsupervised, in conformity with the availability of labels in the target domain. In this study, two unsupervised DA techniques that use deep models are reviewed.

1) ***Deep Models - Unsupervised DA****:* Image analysis progress in the past years is thanks to CNNs, which show to be capable of learning low-level image features. Unfortunately, CNNs also face the domain shift problem.

Image analysis in medicine has one major issue: the lack of labeled data. This is primarily due to the time it takes to annotate images and the money it costs. The field associated with techniques that perform DA when no labeled data in the target domain is available for training is designated unsupervised DA.

A group of techniques that can address unsupervised DA aims to align features, more precisely, to learn domain-invariant features. Figures 4 and 5 illustrate the training and the testing stages of a DA method with invariant feature learning. The objective is to learn features that have the same distribution independently of the domain they belong to. A classifier trained with domain-invariant features in the source domain will most likely perform well in the target domain as the features in which it was trained match the ones in the other domain [11].

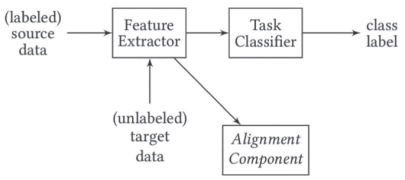

Fig. 4. DA method with invariant feature learning: Training stage [9].

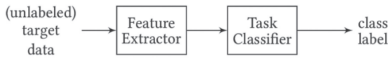

Fig. 5. DA method with invariant feature learning: Testing stage [9].

The architecture presented in figure 4 represents a general description of these training methods. Implementations of this architecture differ in the "Alignment Component" block and in the feature extraction procedure. As shown in figure 4, during the training stage, labeled source data and unlabeled target data enter the "Feature Extractor" block. The extraction can be performed by means of a CNN. The extracted features from the source and target domains enter the "Alignment Component" block to minimize the domain shift. Also, the source features are used to train a task classifier. The source features will be domain-invariant if the alignment is successful, implying that the classifier is trained on domain-invariant features. After the training is complete, the feature extractor can be used on the unlabeled target data to extract relevant features from this dataset; plus, the task classifier can predict the labels for the target data (figure 5). In this study, two strategies for the "Alignment Component" block are investigated, one **minimizes divergence** (Correlation Alignment (CORAL) approach [14]), the other resorts to **adversarial training**, using a domain classifier (Domain-adversarial Neural Network (DANN) approach [15]).

- **CORAL method:** This method aims to align the second-order statistics of the domains by re-coloring whitened source features using the covariance of the target distribution [14]. A whitened feature vector is called this way because it behaves as white noise, having a covariance matrix equal to the identity matrix, meaning that the features of the vector are uncorrelated. When the authors of [14] claim that there is a re-coloring of whitened source features using the covariance of the target distribution, they mean, in other words, that source features which are uncorrelated to other source features (null covariance) will be modified so that they exhibit interdependence with the remaining features present in the feature vector.
  By measuring the distance between the second-order statistics of features extracted from images of the source

and target domains, it is possible to compute the CORAL loss. The classification loss is also computed, being obtained by training the task classifier on the source domain features. During the training the two losses are minimized, the classification loss and the CORAL loss.

- **DANN method:** The feature extractor, together with the task classifier, form a regular feed-forward network. The task classifier receives only the portion of features that correspond to source domain images and predicts their class label. The domain classifier ("Alignment Component" block) receives features extracted from both domains and aims to classify the features as source domain features or as target domain features. If the training intention were to minimize the error associated with the task classifier and the error associated with the domain classifier, then the domain classifier would force the feature extractor to extract dissimilar features across domains. Since the objective is to obtain indistinguishable features while maintaining good performance on the label predictor, a Gradient Reversal Layer (GRL) is introduced, connecting the "Feature Extractor" and the "Alignment Component" blocks. The GRL layer is responsible for the **adversarial training**. During the forward pass, this layer acts as a simple identity transformation. However, during the backward pass, it multiplies the gradients by a negative constant before sending them to the feature extractor, which results in competition between the feature extractor (which tries to induce in error the domain classifier) and the domain classifier (which tries to correctly predict the domain labels).
  In the best-case scenario, the training leads to a domain classifier that achieves 50% accuracy and a task classifier that attains good prediction results. In the end, the feature extractor prevents the domain classifier from correctly predicting the domain labels while still allowing a discriminatory representation for the prediction task. If the domain classifier cannot distinguish between the input features' domains, then the domains are aligned, and the shift is minimized.

III. Methodology

A. Outline

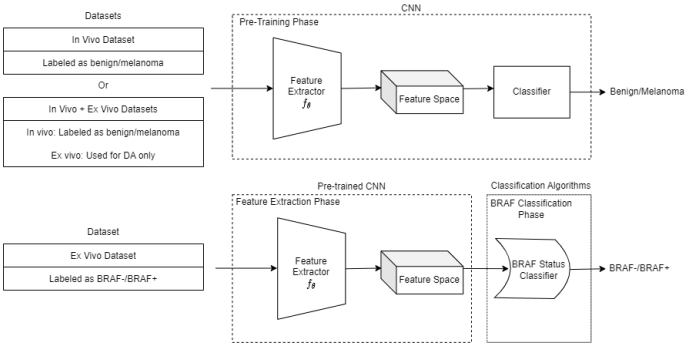

Fig. 6. Proposed pipeline to address BRAF status prediction.

The main objective of this thesis is to predict the mutational status of the BRAF gene in melanoma lesions using faster and more automated methods than the widely used PCR analysis, complementing this procedure. An approach that aims to employ techniques based on DL to analyze dermoscopic images is suggested. The pipeline in figure 6 outlines the proposed approach to tackle the described objective.

Two different dermoscopic datasets are available for this work, one is *in vivo*, and the other is *ex vivo*. The *in vivo* data is vast, and the lesions are labeled as benign/melanoma, while the *ex vivo* images are scarce and correspond to melanomas labeled for the BRAF status and a few benign lesions. In this work, it is intended to predict the mutational status of the BRAF gene. However, the only dataset conveying this information is both small and *ex vivo*. Therefore, there is a need to overcome the few data problem and most likely a domain shift problem, and this is where techniques inspired by FSL and DA will intervene.

As figure 6 outlines, the pipeline is divided in different phases, the "Pre-Training Phase", the "Feature Extraction Phase" and the "BRAF Classification Phase". Each of this phases is now explained in greater detail.

B. Pre-Training Phase

The "Pre-Training Phase" is where the CNNs learn to extract features from different images to obtain knowledge to perform the BRAF classification task. The main goal here is to attain CNNs which can extract relevant information from the input images, i.e., to build a discriminative latent space for the pretended task.

For the "Pre-Training Phase" three strategies are considered: the Related Task (RT) pre-training, the RT & CORAL pre-training and the RT & DANN pre-training.

*1) **RT**:* In this strategy the CNNs are trained for a task which is closely related to the BRAF mutational status prediction task (figure 7). Here, the networks are trained with *in vivo* skin lesion images, learning to classify an image as benign or melanoma. Later on, the knowledge acquired in this task will be used for predicting the BRAF status as there is evidence that some features are related to both tasks.

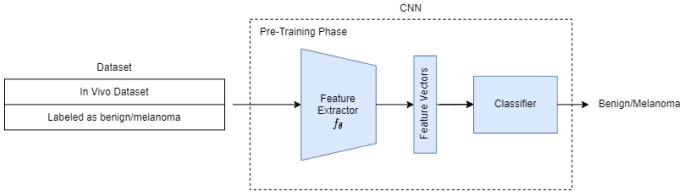

Fig. 7. RT pre-training strategy (● : *In vivo*).

*2) **RT & CORAL**:* Figure 8 illustrates this methodology, which is based on the CORAL loss minimization method proposed by Sun *et al.* [16]. The feature extractor extracts features of images from both domains. The *in vivo* features enter the "Classifier" block to train the model on the RT. The *ex vivo* features enter the "Alignment Component" block together with the *in vivo* features to compute the CORAL

loss. During backpropagation, the training objective is to both minimize the classification loss on the RT and to minimize the CORAL loss, which measures the divergence across the *in vivo* and *ex vivo* domains.

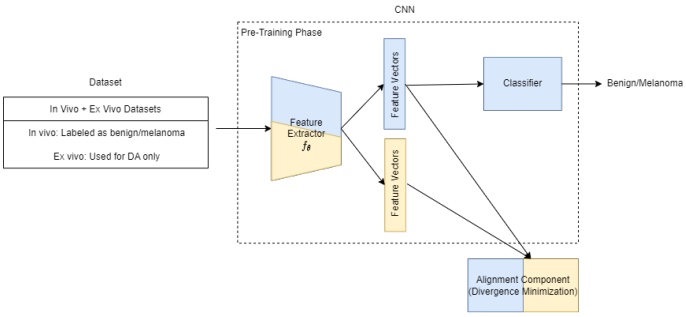

Fig. 8. RT & CORAL pre-training strategy (● : *In vivo*, ● : *Ex vivo*).

3) ***RT & DANN****:* The scheme of this methodology is present in figure 9 and is based on the DANN architecture introduced by Ganin *et al.* [15]. A feature extractor receives the *in-vivo* and the *ex-vivo* data. The output of this extractor is a set of feature vectors for the *in vivo* data and another for the *ex vivo* data. The extracted data must be, first of all discriminative enough to perform the RT and secondly needs to be domain-invariant. To achieve both goals, at the output of the feature extractor there is a bifurcation, one path leads the *in vivo* features to the RT classifier while the other path leads both *in vivo* and *ex vivo* features to the "Alignment Component" block which consists of a domain classifier. The domain classifier is trained to distinguish the two domains, and the benign/melanoma classifier block ("Label Predictor") is trained to minimize the classification error on the *in vivo* data. A GRL connecting the "Feature Extractor" and the "Alignment Component" blocks ensures that the adversarial training takes place. The feature extractor is forced to learn domain-invariant but discriminative features that can trick the domain classifier (minimizing the shift) and produce correct classifications for the RT at the same time.

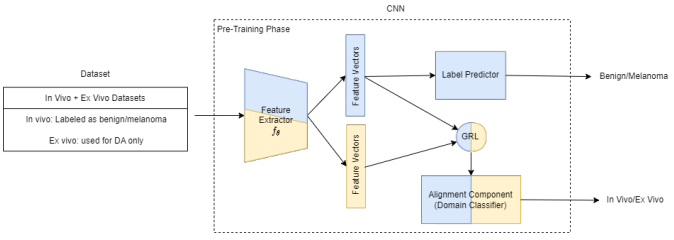

Fig. 9. RT & DANN pre-training strategy (● : *In vivo*, ● : *Ex vivo*).

C. Feature Extraction and BRAF Classification Phases

After the pre-training stage, the networks can be used as feature extractors on the *ex vivo* data labeled for the BRAF status ("Feature Extraction Phase"). The extracted features are then processed by external classification algorithms during "BRAF Classification Phase". Figure 10 illustrates this procedure.

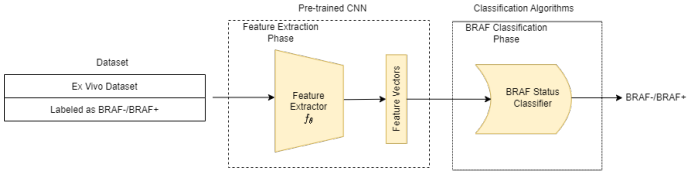

Fig. 10. Feature extraction & BRAF status classification (● : *Ex vivo*)

1) ***K-Nearest Neighbors****:* The K-Nearest Neighbors (KNN) algorithm [17] is one of the simplest algorithms used for classification. The idea is to compare a test sample (test feature extracted from *ex vivo* data) to the training samples (training features also extracted from the *ex vivo* dataset). The label assigned to the test example will be the same as the most prevalent class of the nearest k training examples. In this research the euclidean distance and the cosine similarity are the considered distance metrics.

2) ***Prototype-based Classifiers****:* The prototype-based classifiers are inspired on the metric-based FSL classification algorithms present on the MatchingNet and on the ProtoNet.

- **ProtoNet Classifier With Euclidean Distance (P1)** is the classifier proposed in [8]. In this thesis, the feature extractor extracts the features from the *ex vivo* images. After this, the BRAF- and the BRAF+ features are gathered into two different groups. By averaging the features of each group, two prototypes are computed, one for the BRAF- class and another for the BRAF+ class. The euclidean distance between a test feature and each class prototype is calculated to predict the class for the test example. The label assigned to the sample is given according to the spatially closer prototype;
- **ProtoNet Classifier With Cosine Similarity (P2)** is similar to P1, the only difference is that instead of using the euclidean distance as the comparison metric, here, the cosine similarity between test samples and each class prototype is computed;
- **MatchingNet Classifier With Cosine Similarity (M1)** is the standard classifier proposed by Vinyals *et al.* [7]. In this strategy, after the features of the *ex vivo* dataset are extracted and divided into two different groups, the BRAF- and the BRAF+ group, the cosine similarity between a test example and each one of the samples in the two groups is computed. This leads to N similarity values with respect to the BRAF+ class and M similarity values with respect to the BRAF- class. To achieve a single similarity value per class, the average cosine similarities are computed by averaging the similarity values of each class group. In the end, the class assigned to the test example is given according to the highest average similarity value;
- **MatchingNet Classifier With Euclidean Distance (M2)** is similar to M1, but instead of computing the average cosine similarity per class, now the average euclidean distance per class is determined.

3) ***Logistic Regression****:* A Logistic Regression (LR) [18] is trained on the extracted features from the *ex vivo* data. A test sample can then be fed to the LR equation to predict the

mutational status of the patient associated with the extracted features. In this work, the LR is used to model the a posteriori probabilites of the BRAF+ and BRAF- classes. Besides the classical LR model, others are considered, namely LR with a regularization parameter. The regularizers contemplated for this strategy are the $L1$ norm ($L1$ penalty) and the $L2$ norm ($L2$ penalty).

IV. Experimental Set-Up

A. Datasets

1) ***Analysis****:* For this thesis, dermoscopic images from different data centers are available. The largest dataset is public and consists of *in vivo* skin lesions (International Skin Imaging Challenge (ISIC) 2020 dataset [19]). This dataset comprehends $33,126$ *in vivo* dermoscopic images of skin lesions, of which $32,542$ are benign lesions and 584 are melanomas. The images come from six different centers. Thus, this dataset presents multi-source data and the images have different colors, sizes, and aspect ratios.

The smallest dataset is private and contains 138 *ex vivo* dermoscopic images of skin lesions. Of these lesions, only 69 are melanomas and labeled for BRAF status. This portion of the dataset will be denoted the "BRAF dataset". The remaining 69 images from the 138 are benign and unlabeled for BRAF status. The 69 images that comprehend the BRAF dataset correspond to 43 melanoma patients, meaning that patients may present more than one image for the same lesion, depending on its size. Of the 43 patients, 23 are men (53.49%), and 20 are women (46.51%). The average age of the patients is 69.60 ± 11.45 years old. This dataset is also imbalanced because there are 31 BRAF- patients (48 images) and 12 BRAF+ patients (21 images).

The private *ex vivo* dataset is used for two purposes. The first purpose is to perform BRAF status classification, and for this, only the BRAF dataset portion is considered. The other purpose is to perform DA; to do so, the total amount of 138 images is taken into account.

2) ***Pre-Processing****:* From the $33,126$ images of the ISIC 2020 dataset, 425 are duplicates. The duplicates are discarded and the remaining data is split into two different sets. A training set (80% of the whole dataset) and a validation set (20% of the whole dataset). The images in the *in vivo* and *ex vivo* datasets are resized to a 300×300 resolution and padded with two horizontal black borders to preserve the original aspect ratio and the information. Additionally, since the *in vivo* data comes from different centers, which operate under different illumination conditions and use different acquisition devices, the color constancy algorithm Shades of Gray [20] is applied to the data.

B. Evaluation Metrics and Implementation Challenges

1) ***Metrics****:* To evaluate the different models used in this work, the following metrics are considered: Specificity (SP), Sensivity (SE), Balanced Accuracy (BACC), Precision (PR), and F_1 score (F_1).

2) ***Generalization Problem****:* The few available data to perform the BRAF status classification is a major issue. To train the proposed BRAF classification algorithms that act on the information extracted by the CNNs, a leave-one-out cross-validation [21] is applied. This way, the obtained results are more robust. The algorithms are trained with the information extracted from 42 of the 43 BRAF patients and validated on the case that is left out.

3) ***Multiple Images Per Patient Problem****:* Given that on the BRAF dataset, a patient can present more than one image for the same lesion, two strategies are proposed to address this issue. In the first, the models are trained using all the images and then the classification information is gathered (either assuming that a single BRAF+ prediction is enough to classify the patient as so, One-Dominance (1D), or using Majority Voting (MV)). The second strategy is a Summarized Features (SF) analysis, where after performing feature extraction on every image for a patient, the information is gathered in a summary vector, either using the mean value (Mean) or the max-wise value (Max) across all of the feature vectors' values.

C. Pre-Training Approaches and Computational Environment

Three different CNN architectures are chosen to perform the feature extraction procedure. The selected architectures are the ResNet-18 [22], the EfficientNet-B2 [23], and the Inception-V3 [24].

Regarding the pre-training approaches, all of them use on-line data augmentation consisting of random horizontal and vertical flips as well as a random erasing with a probability of 50% (scale = $(0.02, 0.33)$, ratio = $(0.3, 3.3)$) in the pre-training data. Besides, the feature extractor is initialized in the ImageNet [25] classification task and the last fully-connected layer is replaced by a randomly initialized lesion classifier with the same number of input units as the extracted features' size and two output units. This lesion classifier introduces a dropout with probability $p = 0.3$ (for the ResNet-18 and the EfficientNet-B2 architectures). As for the lesion classification loss, categorical cross-entropy with 0.02 penalty for misclassifications in the benign class and 0.98 penalty for misclassifications in the melanoma class is considered. Finally, The optimizer considered during training is the Adam optimizer [26], and the initial learning rate is 5×10^{-7} for the ResNet-18 and 1×10^{-6} for the Inception-V3 and the EfficientNet-B2.

As for the configurations which are more specific to each pre-training approach, the RT pre-training is set for 100 epochs with an early-stop of patience = 30 and uses a batch-size of 32 for the ResNet-18 and Inception-V3 architectures and 16 for the EfficientNet-B2. The RT & CORAL pre-training is set for 100 epochs controlled by the *in vivo* dataset's size and the CORAL loss with a lambda value of 100 is chosen for the domain loss. Regarding the batch-size, batches of 64 images are considered for the ResNet-18, 32 for the Inception-V3, and 20 for the EfficientNet-B2. Lastly, the RT & DANN pre-training is set for 100 epochs controlled by the *in vivo* dataset's size and the domain classifier consists in a MLP composed of three layers. The input layer and the hidden layer

present, at the input and at the output, the same number of units. This number of units equals the number of features in a feature vector. Between these two layers there is a dropout with probability $p = 0.3$. The output layer has an input size equal to the feature vectors' dimension and contains one output unit. The domain classification loss is the binary-cross entropy and the selected batch sizes are 64 for the ResNet-18 and 16 for the other architectures.

All of the experiments in this thesis were conducted using the Python programming language. The framework used to manipulate DL architectures was the $1.12.1$ version of Pytorch [27]. Other Python libraries such as Sklearn and Numpy were also frequently used to assess classification performance. The training of DL architectures was performed on a desktop with an Intel(R) Core(TM) i7-7700 CPU @ 3.60GHz, 3601 Mhz, 4 Core(s), 8 Logical Processor(s) and a GeForce GTX 1060 6GB NVIDIA GPU.

V. Experimental Results and Discussion

A. Best Results for BRAF Status Prediction and Comparison with the State of the Art

Table I contemplates the result of applying the predictor of [4] to the available BRAF dataset. Also, it summarizes the best results attained for BRAF status classification using the ResNet-18 and the EfficientNet-B2 as feature extractors. The results for the Inception-V3 architecture were similar to the ones obtained for the EfficientNet-B2 both for BRAF status prediction and for the RT. Therefore, they are not presented. Besides, during the pre-training stage, for all approaches, the Inception-V3 architecture has exhibited convergence issues, experiencing a performance drop very early during training compared to the ResNet-18 and the EfficentNet-B2 architectures.

It can be seen that the decision tree proposed by Armengot-Carbó *et al.* [4] achieves a high SP value (77.4%). However, the SE value is low (around 16.7%). The tree classifier does not lead to the best results. The reason behind this may be related to the age distribution of the BRAF dataset (69.60 ± 11.45), which is different from the one of the dataset available in [4]. Besides, for the application of the decision tree classifier, the melanomas located in the palmoplantar and facial regions were not discarded, like in [4], since in this thesis, the BRAF dataset was already small.

The best results were obtained using the EfficientNet-B2 as extractor and the LR with L1 penalty as BRAF classification algorithm, as a matter of fact, this approach surpasses the state-of-the-art algorithm in every aspect. For this architecture, the different proposed pre-training approaches did not benefit the BRAF classification performance since the best results were obtained by initializing the architecture with the weights resultant from a pre-training on the ImageNet dataset.

The ResNet-18 architecture, unlike the EfficientNet-B2, benefited from the proposed pre-training approaches. The best results for this architecture were obtained by considering the RT pre-training approach and a prototype-based classifier (P2 classifier).

Despite the data limitation, it is possible to achieve reasonable results for the BRAF status prediction. Furthermore, it is interesting to note that for the two architectures, the selected BRAF status classifier is the same in the per-image and SF analyses. As for the preferred network pre-training, there may be several reasons why one architecture does not benefit from the pre-training approaches, and the other one does. The most plausible one has to do with the fact that the EfficientNet-B2 and the ResNet-18 architectures come from different families, each with different building principles.

In sum, comparing the best results obtained in this investigation with the results obtained by employing the classifier developed in [4], it can be concluded that the proposed approaches are viable, achieving better results than the ones obtained through the algorithm presented in the state of the art [4].

B. Performance on the RT

The results obtained on the RT, for the different pre-training approaches (RT, RT & CORAL, and RT & DANN) for the ResNet-18 architecture can be observed in table II. The results obtained for the EfficientNet-B2 were similar and are omitted for the sake of simplicity.

TABLE II
ResNet-18 performance on the RT. For the *in vivo* performance, the networks were evaluated on the **ISIC 2020 validation set** ($6,196$ images: $6,079$ benign, 117 melanomas), whereas for the *ex vivo* performance, the networks were evaluated on the **private *ex vivo* dataset** (138 images: 69 benign, 69 melanomas).

***In Vivo* Performance**					
Pre-Train	Metrics				
	BACC(%)	SP(%)	SE(%)	F1(%)	PR(%)
RT	**75.6**	**87.9**	**63.3**	**15.9**	**9.1**
RT & CORAL	73.6	93.3	53.9	21.4	13.4
RT & DANN	73.1	91.6	54.7	18.4	11.1
***Ex Vivo* Performance**					
Pre-Train	Metrics				
	BACC(%)	SP(%)	SE(%)	F1(%)	PR(%)
RT	54.4	10.1	98.6	68.3	52.3
RT & CORAL	**59.4**	**52.2**	**66.7**	**62.2**	**58.2**
RT & DANN	51.5	98.6	4.4	8.2	75.0

The best performance on the benign/melanoma classification task for the *in vivo* dataset is obtained by the simple RT pre-training. However, when it comes to the private *ex vivo* dataset, the pre-training that adds CORAL loss minimization between the *in vivo* and *ex vivo* domains (RT & CORAL) attains the best performance, costing only a small performance drop on the *in vivo* data when compared to the RT pre-training case. For the *ex vivo* data, the RT & CORAL pre-trained network exhibits balanced values of SP and SE, contrary to what happens in the two other pre-training approaches where there is either a high SE value and a low SP value (on the RT case) or a high SP value and a low SE value (on the RT & DANN case).

In the RT & DANN pre-training approach, the high SP and low SE values obtained for the *ex vivo* data may be due to the alignment of *ex vivo* melanomas with *in vivo* benign lesions, as the gross part of the *in vivo* data consists of benign lesions.

TABLE I
STATE OF THE ART COMPARED TO THE BEST RESULTS PER ARCHITECTURE FOR BRAF STATUS PREDICTION

Medical Analysis						
Classifier	Pre-Trained Network	Metrics				
		BACC(%)	SP(%)	SE(%)	F1(%)	PR(%)
Decision Tree [4]	-	47.1	77.4	16.7	19.1	22.2
Per-Image Analysis						
Classifier	Pre-Trained Network	Metrics				
		BACC(%)	SP(%)	SE(%)	F1(%)	PR(%)
LR_{L1} - 1D	EfficientNet-B2: ImageNet	75.3	83.9	66.7	64.0	61.5
P2 - MV	ResNet-18: RT	62.1	74.2	50.0	46.2	42.9
SF Analysis						
Classifier	Pre-Trained Network	Metrics				
		BACC(%)	SP(%)	SE(%)	F1(%)	PR(%)
LR_{L1} - Max	EfficientNet-B2: ImageNet	77.8	80.7	75.0	66.7	60.0
P2 - Max	ResNet-18: RT	60.5	71.0	50.0	44.4	40.0

C. BRAF Classification Performance

The EfficientNet-B2 architecture initialized with the ImageNet weights led to better results on the BRAF classification task than models pre-trained on RT. Contrarily to the EfficientNet-B2, the ResNet-18 architecture benefited from the proposed pre-training approaches. Therefore, the best results for BRAF status classification, for each pre-training approach, achieved using this CNN architecture as a feature extractor are analyzed in this section.

*1) **KNN**:* Experiments on the extracted features were conducted considering the euclidean distance (Euc.) and the cosine similarity (Cos.). The number of neighbors considered in these experiments was either 3 (3N) or 5 (5N). The results obtained for the per-image analysis are shown in figure 11, and for the SF analysis, in figure 12.

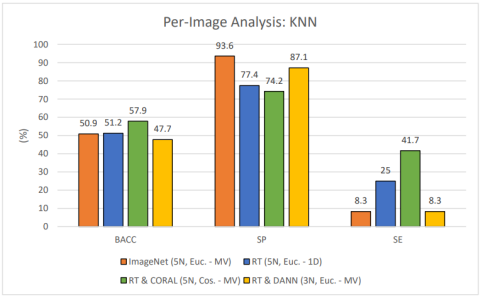

Fig. 11. Per-Image analysis: **BACC**, **SP**, and **SE** values for the BRAF status classification using pre-trained **ResNet-18** architectures as feature extractors on the BRAF dataset and the **KNN algorithm**.

From figures 11 and 12, considering the BACC scores, one can conclude that the best result for the per-image analysis is attained when considering the RT & CORAL pre-training approach for the feature extractor, the KNN classifier configured with 5N, cosine similarity, and the MV criterion. The best SF analysis result is attained when considering the pre-training on the RT for the extractor, the KNN classifier configured with 3N, cosine similarity, and the Max criterion. Despite the best pre-training approach not being the same in the two types of analyses, the selected distance metric for the KNN is equal.

In general, the KNN classifier leads to biased results, exhibiting high SP and low SE values in both analyses.

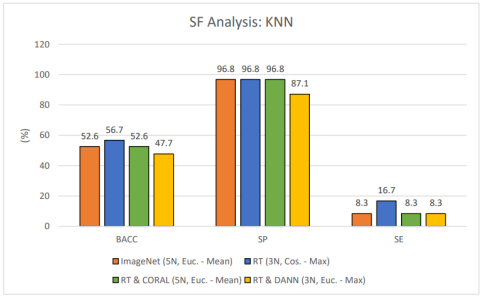

Fig. 12. SF analysis: **BACC**, **SP**, and **SE** values for the BRAF status classification using pre-trained **ResNet-18** architectures as feature extractors on the BRAF dataset and the **KNN algorithm**.

*2) **Prototypes**:* For the prototype-based classifiers, the P1, P2, M1, and M2 classifiers introduced in chapter III were applied to the extracted feature vectors. The results obtained for the per-image analysis are presented in figure 13 and for the SF analysis in figure 14.

For the prototypes approach, it is evident that there is a preference for the MV criterion in the per-image analysis and a preference for the Max criterion in the SF analysis.

The RT pre-training put together with the P2 BRAF classifier achieves the highest BACC scores (62.1% for the per-image analysis and 60.5% for the SF analysis); besides, it displays balanced SE and SP values in both scenarios.

The RT & DANN pre-trained extractor also presents acceptable results in the two types of analyses despite being a slightly worse option when compared to the architecture initialized with the ImageNet weights for the SF analysis, which exhibits a better SP value. The RT & CORAL pre-trained ResNet-18 architecture exhibits the worse results for the prototype-based classifiers, manifesting a BACC value lower than 50% in the per-image and the SF analyses.

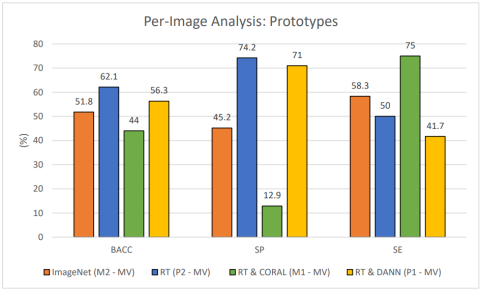

Fig. 13. Per-image analysis: **BACC**, **SP**, and **SE** values for the BRAF status classification using pre-trained **ResNet-18** architectures as feature extractors on the BRAF dataset and the **prototype-based algorithms**.

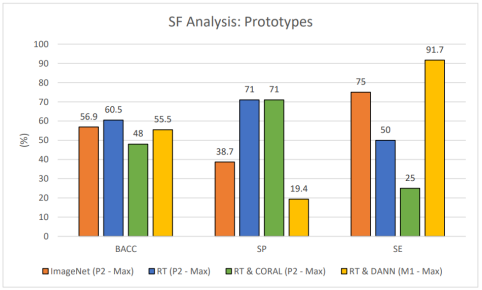

Fig. 14. SF analysis: **BACC**, **SP**, and **SE** values for the BRAF status classification using pre-trained **ResNet-18** architectures as feature extractors on the BRAF dataset and the **prototype-based algorithms**.

D. LR

The LR classifier was trained on the features extracted from the BRAF dataset. Three configurations for the LR were considered: LR with $L1$ penalty to compute sparse solutions when existent, reducing irrelevant coefficients to zero, i.e., removing unnecessary features for the BRAF classification, LR with $L2$ penalty, and classical LR with no regularization terms.

Figure 15 shows that there is an unanimous choice in terms of per-image analysis criterion since all the classifiers configurations present the best results when MV is considered. As exhibited in figures 15 and 16, for all the pre-training approaches, the LR configuration for the per-image analysis coincides with the configuration selected for the SF analysis. Moreover, for all the pre-training approaches, the regularized versions of the LR attained the best results.

The feature extractor pre-trained on the RT and the feature extractor pre-trained on the RT & CORAL exhibit the best results when the $L2$ penalty LR is considered. However, the networks pre-trained on the ImageNet dataset and on the RT & DANN approach attain the best results when the $L1$ penalty term is taken into account instead. This said, there is a high chance that the feature vectors extracted using these last two mentioned pre-training approaches contain information that is irrelevant for BRAF classification and when this information is discarded, the models perform well for this task. In fact, the RT & DANN pre-trained network used as feature extractor together with the LR classifier with the $L1$ penalty leads to a BACC of 57.9% on the per-image analysis and a BACC of 59.5% in the SF analysis, the best results achieved in these analyses with the LR classifier.

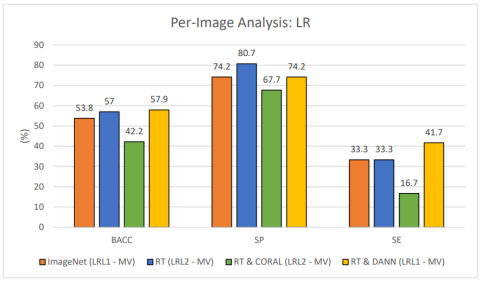

Fig. 15. Per-image analysis: **BACC**, **SP**, and **SE** values for the BRAF status classification using pre-trained **ResNet-18** architectures as feature extractors on the BRAF dataset and the **LR**.

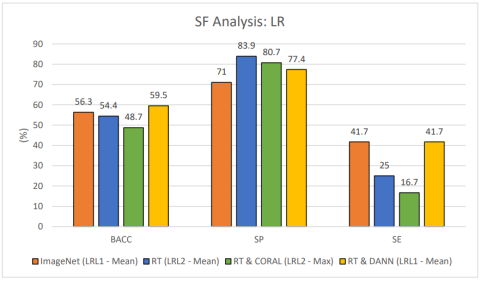

Fig. 16. SF analysis: **BACC**, **SP**, and **SE** values for the BRAF status classification using pre-trained **ResNet-18** architectures as feature extractors on the BRAF dataset and the **LR**.

E. Final Considerations on BRAF Classification

From the conducted experiments with the ResNet-18 as a feature extractor, it is observed that the KNN classifier is not the best option for classifying BRAF status. The BRAF dataset presents more BRAF- samples than BRAF+ samples, and the achieved results show that the KNN algorithm presents, in a general manner, tendency to classify the samples as BRAF-, being susceptible to imbalanced data. The results obtained exhibit high SP values and low SE values, making this classifier biased for all the pre-training approaches.

The prototype-based classifiers show promising results, especially the classifiers inspired by the ProtoNet (P1 and P2). Comparing the prototypes to the KNN and the LR classifiers, the prototype-based P2 classifier, together with the RT pre-training, achieved not only the most balanced results in terms of SE and SP but also the best results in terms of BACC for the per-image and the SF analyses. The LR classifier also achieved reasonable results, despite being less balanced in terms of SE and SP than the results obtained using the prototype-based classifiers.

Finally, it can be concluded that when a patient presents more than one image per lesion, it is possible to summarize the information in the multiple feature vectors and still achieve results similar to those obtained through the per-image analysis. In fact, the results achieved for the three families of classifiers show that in both types of analyses, there is a tendency for the best classifier configuration to be equal.

VI. Conclusions and Further Investigation

*1) **Conclusions**:* The findings of this study prove that resorting to DL methods to analyze dermoscopic data for BRAF status prediction can surpass the existent state-of-the-art results (algorithm proposed in [4]).

The leave-one-out cross-validation [21] proved to be a reliable approach in this study to evaluate the models, given the lack of data problem. Plus, the two types of analyses selected to deal with the multiple images per patient problem (the per-image and the SF analyses) tend to lead to the same behaviors for the ResNet-18 architecture, showing that it is possible to summarize the information for a patient presenting multiple data for the same lesion in a single information vector.

Unfortunately, the Inception-V3 and the EfficientNet-B2 architectures did not benefit from the proposed pre-training approaches for the BRAF status prediction task. So, the information learned by the feature extractor during pre-training is highly dependent on the selected architecture.

Regarding the classifiers adopted for this work, the prototype-based classifiers emerge as promising classification algorithms for their good performance on the BRAF classification task and simplicity.

*2) **Further Investigation**:* This thesis foments further research around BRAF status prediction using computer-aided methods. The results obtained for BRAF status prediction are promising. However, this work still has room for improvement. One limitation of the present work is that it does not explain the relation between the information present in the dermoscopic images and the BRAF status. Nevertheless, there are some ideas to study the aforementioned relation and to improve the work in this thesis: **i)** It would be interesting to visualize the activation maps of the CNNs to further understand what patterns they learn and why some architectures benefit from pre-training tasks closely related to the BRAF detection problem and some don't. Plus, it would also be interesting to correlate the patterns learned with the information retrieved by dermatologists; **ii)** Instead of just aligning the *ex vivo* data with the *in vivo* data, a total alignment between the different data centers of the ISIC 2020 dataset and the *ex vivo* data should be explored to check if there is any benefit following a total DA strategy; **iii)** To increase the reliability of the proposed BRAF classification algorithms, new dermoscopic data should be used to test the algorithms. Other types of medical images like WSIs, which convey information about cellular morphology, could also be used to further study the proposed methodologies.

References

[1] L. E. Davis, S. C. Shalin, and A. J. Tackett, "Current state of melanoma diagnosis and treatment," *Cancer Biology & Therapy*, vol. 20, no. 11, pp. 1366–1379, 2019.

[2] The American Cancer Society medical and editorial content team. Targeted Therapy Drugs for Melanoma Skin Cancer. Accessed 19-November-2021. [Online]. Available: https://www.cancer.org/cancer/melanoma-skin-cancer/treating/targeted-therapy.html

[3] T. J. Brinker *et al.*, "Deep learning approach to predict sentinel lymph node status directly from routine histology of primary melanoma tumours," *European Journal of Cancer*, vol. 154, pp. 227–234, 2021.

[4] M. Armengot-Carbó *et al.*, "The association between dermoscopic features and BRAF mutational status in cutaneous melanoma: Significance of the blue-white veil," *Journal of the American Academy of Dermatology*, vol. 78, no. 5, pp. 920–926, 2018.

[5] Y. Wang *et al.*, "Generalizing from a Few Examples: A Survey on Few-Shot Learning," *ACM Comput. Surv.*, vol. 53, no. 63, pp. 1–34, 2020.

[6] W. Zi, L. S. Ghoraie, and S. Prince. Tutorial # 2: few-shot learning and meta-learning I. Accessed 19-January-2022. [Online]. Available: https://www.borealisai.com/en/blog/tutorial-2-few-shot-learning-and-meta-learning-i/

[7] O. Vinyals *et al.*, "Matching Networks for One Shot Learning," in *Proceedings of the 30th International Conference on Neural Information Processing Systems*, 2016, pp. 3637–3645.

[8] J. Snell, K. Swersky, and R. Zemel, "Prototypical Networks for Few-shot Learning," in *Advances in Neural Information Processing Systems*, 2017.

[9] W.-Y. Chen *et al.*, "A Closer Look at Few-shot Classification," in *International Conference on Learning Representations*, 2019.

[10] H. Guan and M. Liu, "Domain Adaptation for Medical Image Analysis: A Survey," *IEEE Transactions on Biomedical Engineering*, vol. 69, no. 3, pp. 1173–1185, 2021.

[11] G. Wilson and D. J. Cook, "A Survey of Unsupervised Deep Domain Adaptation," *ACM Transactions on Intelligent Systems and Technology*, vol. 11, no. 51, pp. 1–46, 2020.

[12] S. J. Pan and Q. Yang, "A Survey on Transfer Learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[13] H. Daumé and D. Marcu, "Domain Adaptation for Statistical Classifiers," *J. Artif. Int. Res.*, vol. 26, pp. 101–126, 2006.

[14] B. Sun, J. Feng, and K. Saenko, "Return of Frustratingly Easy Domain Adaptation," in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, 2016, pp. 2058–2065.

[15] Y. Ganin *et al.*, "Domain-Adversarial Training of Neural Networks," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 2030–2096, 2016.

[16] B. Sun and K. Saenko, "Deep CORAL: Correlation Alignment for Deep Domain Adaptation," in *Computer Vision – ECCV 2016 Workshops*, 2016, pp. 443–450.

[17] T. M. Mitchell, *Machine learning*. McGraw-hill New York, 1997.

[18] C.-Y. J. Peng, K. L. Lee, and G. M. Ingersoll, "An Introduction to Logistic Regression Analysis and Reporting," *The journal of educational research*, vol. 96, no. 1, pp. 3–14, 2002.

[19] V. Rotemberg *et al.*, "A Patient-Centric Dataset of Images and Metadata for Identifying Melanomas Using Clinical Context," *Scientific Data*, vol. 8, no. 34, pp. 1–8, 2021.

[20] G. D. Finlayson and E. Trezzi, "Shades of Gray and Colour Constancy," in *Proc. 12th Color Imag. Conf.: Color Sci. Eng. Syst., Technol., Appl.*, 2004, pp. 37–41.

[21] "Leave-One-Out Cross-Validation," in *Encyclopedia of Machine Learning*, C. Sammut and G. I. Webb, Eds., 2010, pp. 600–601.

[22] K. He *et al.*, "Deep Residual Learning for Image Recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[23] M. Tan and Q. V. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," in *International conference on machine learning*, 2019, pp. 6105–6114.

[24] C. Szegedy *et al.*, "Rethinking the Inception Architecture for Computer Vision," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 2818–2826.

[25] J. Deng *et al.*, "ImageNet: A Large-Scale Hierarchical Image Database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[26] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," 2014. [Online]. Available: https://arxiv.org/abs/1412.6980

[27] A. Paszke *et al.*, "PyTorch: An Imperative Style, High-Performance Deep Learning Library," in *Advances in Neural Information Processing Systems*, 2019, pp. 8024–8035.