Machine Learning for Automated Diagnosis of Skin Lesions

Fábio Santos, Email: fmts@ua.pt
Supervisors: Filipe Silva, Pétia Georgieva
Department of Electronics Telecomunications and Informatics
University of Aveiro, Aveiro, 3810-193

Abstract—Machine learning, specifically, deep learning is a fast-growing field that is being used for multiple medical imaging related problems, such as early detection of skin cancer. For a long time, automated diagnosis of skin cancer from clinical images was considered to be out of reach. However, recent works based on deep networks produced promising results which have the potential to change the landscape of skin lesion diagnosis. Systems created based on these new advancements aim to provide support for both dermatologists in the decision making process and for patients that do not have access to skin professionals. This paper focuses on the current state of automated skin lesion diagnosis using convolutional neural network models, while also providing a comprehensive look into the requirements of integrating such models into a web application capable of helping dermatologists on the diagnosis of skin lesions.

Index Terms—Automatic assessment tools, eHealth, machine learning, medical imaging

1 Introduction

1.1 Background

Skin cancer is the most common type of cancer, particularly, in the United States of America the incidence rates keep rising with currently 1 in 5 persons developing skin cancer until the age of 70 [1]. However, skin cancer represents a problem not only for America but also for the international health community in general. For example, in Europe, over 100000 people are diagnosed with melanoma and 22000 deaths annually occur due to this form of skin cancer [2]. Yet, one of the most remarkable facts about skin cancer is that when detected on late stages there is a 23% chance of survival, but when detected early the 5 year survival rate rises to 99% [1]. Therefore, the early detection of skin cancer is an absolute priority.

Skin cancer can be detected by dermatology professionals by simple visual examination of skin lesions. However, the difference between malignant and benign skin lesions can be negligible making it a difficult task even for trained medical experts. As such, a medical application which provides automated skin lesion diagnosis for decision support is an welcome addition to this field.

Initially, automated diagnosis of skin lesions was made based on predefined techniques well known by dermatology professionals such as the ABCDE rule (Asymmetry, Border, Color, Dermoscopic structure and Evolving), but often failed to either generalize to new cases or lacked the accuracy of a human. However, in more recent years, machine learning approaches into skin lesion diagnosis shows remarkable performance in comparison with the hand crafted algorithms, specially with deep learning methods [3] [4].

1.2 Objectives and Motivation

A strong assumption can be made that if an accurate automated skin lesion diagnosis tool is used by dermatologists, then skin cancer cases will be detected earlier. Therefore, the main objective behind this dissertation is to improve the current work on automated skin lesion diagnosis by using deep learning techniques. This work will be part of a tool that has the intent of being used in a clinic context, so the priority is to its performance as much as possible.

Finally, the aforementioned tool must be packaged within a eHealth application, in order to be easily accessed by medical professionals in a clinical environment. Such application must have the ability to potentially integrate other components useful for both dermatologists and patients while enabling easy communication between them.

2 Literature review

The following sections are structured as follows. Section 2.1 provides an exploratory look into eHealth/mHealth applications for skin lesion diagnosis. Section 2.2 gives an overview of deep learning applied into medical imaging. Finally, Section 2.3 reviews some of the current approaches towards skin lesion diagnosis systems using deep learning.

2.1 eHealth/mHealth for skin lesion diagnosis

Online health, ehealth and mhealth applications represent a rapidly developing field of medicine that has the potential to become powerful tool in the diagnosis and management of skin diseases [5]. These applications aim to enhance clinical care, promote health, prevent diseases and most importantly provide medical support when it is not available at a particular location or time. Generally, the acceptance towards this type of systems in the medical community keeps growing, but is highly dependent on factors such as performance, accessibility and ease of use, which poses challenges for their global adoption.

Currently, several production ready skin lesion classification systems are available for both skin professionals and patients wishing to self monitor their own skin. However, almost none of them has shown to be sufficiently accurate or reliable enough for a clinical environment.

One of the most popular eHealth applications for this purpose is Metaoptima's Dermengine web application. Their Visual Search tool compares a user-submitted image with similar images in a database of thousands of pathology-labelled images gathered from other dermatologists. Deep learning techniques are used to search for related images based on visual features such as colour, shape or patterns [6].

Another popular app is the SkinVision which classifies lesions as either low, medium or high risk of skin cancer by using an risk assessment algorithm based on grayscale images of lesions and their associated fractal maps. It achieves the overall sensitivity of 73%, specificity of 83%, and accuracy of 81%. The positive and negative predictive values were 49% and 83%, respectively [5].

2.2 Deep neural networks for medical imaging

Deep learning refers to computational models composed of multiple processing layers capable of learning representations of data with multiple levels of abstraction [7]. The initial impact of deep learning for medical imaging was revealed through a special issue published in 2016 at the IEEE Transactions on Medical Imaging [8]. It explains the principles and methods of deep learning applied to medical image analysis. These structures can be found in approaches to medical imaging problems such as organ segmentation, lesion detection and tumor classification.

The main advantage of deep learning over other machine learning algorithms is that it removes the need for feature engineering, a process that requires knowledge of the problem domain, which can be a time consuming process as well as introduce human error.

Recently, deep neural networks appear as state-of-the-art solutions for medical imaging problems due to advancements in the field. These advancements include the research and development of new methods to prevent overfitting, the rise of computational power along with the use of graphical processing units, and finally the development of high level modules such as theano [9] that help train and test neural networks .

2.3 Skin lesion classification using deep learning

One of the most important factors which determines the performance of a deep learning model is the dataset used to train it on. For general image recognition problems datasets such as ImageNet [10] which contains over 14 Million samples with over 20000 classes are usually used and serve as a benchmark. However for skin lesion diagnosis systems, it is difficult and in many times impossible to compare the performance of published classification results since many authors use nonpublic datasets for training and testing [11]. The closest benchmark available for this domain is the HAM10000 dataset [12].

Nonetheless, organizations such as International Skin Imaging Collaboration (ISIC) provide an open source public access archives of skin images, which can be used for teaching or for the development and testing of automated skin lesion diagnosis systems [13]. They also place a challenge around their dataset every year in order to improve the performance of this classification systems as a whole.

2.3.1 *Transfer learning approaches*

The most popular approach for skin cancer classification using deep learning, specifically, convolutional neural networks, was published in 2017 by Esteva et al. [3] and could diagnose keratinocyte and melanoma cancer. The authors follow a transfer learning approach by leveraging the weights of the InceptionV3 network trained on ImageNet, on top of which they build their own classifier. Finally, they measured the network's performance by pitting it against 21 dermatologists and concluded that their classifier had comparable performance to that of those board-certified dermatologists. This network used a very large set of labelled images in order to achieve high accuracy, namely, 129450 clinical images. This data was a combination of multiple sources, both proprietary and publicly available.

In the meantime, submissions to ISIC challenges have also been trying new concepts that improve the skin lesion diagnosis model's performance. In the part 3 of the ISIC 2018 challenge participants were asked to develop a classifier to distinguish between 7 different types of skin cancer and the ranking was made based on their normalized multiclass accuracy [14]. The top 3 submissions had balanced accuracies of about 88,5%, 88,2%, 87,1% respectively and were all submited by Metaoptima (the company behind Dermengine) [15]. To train those models they used the provided dataset along with proprietary data. Additionally, they augmented the training data by performing random horizontal flips, random rotations, changes in brightness, saturation, and contrast. They used transfer learning from several pre trained models trained on ImageNet (such as InceptionV3 or ResNet) and then ensembled the best performing ones [15].

The 2019's version of this challenge asked participants to classify dermoscopic images among nine different diagnostic categories, however this time around one of the classes was "unknown" (none of the others). Similarly to the 2018's version participants could use their own data to improve the network's performance and were ranked based on a balanced multiclass accuracy [13]. The results turned out to be quite promising, with the best submission posted by Geesert et al. [16] scoring 92.6% accuracy. They trained their networks on a combination of multiple datasets that also included the HAM10000 [12].

Gessert's et al. approach to preprocessing was to first crop the images, perform image binarization, apply the shades of gray color constancy method and finally resize the images. Data augmentation is also applied by randomly changing brightness, contrast, rotation, scale, shear and flip. They used two different input strategies, the first takes a random crop from the preprocessed image, the second randomly resizes and scales the image when taking a crop from the preprocessed one. Like the previous attempts they use a transfer learning approach relying on EfficientNets that were trained on the ImageNet dataset. For each model, predictions for each model are made based on which input strategy was used. The final prediction is made using an ensemble of a subset which contained all the best performing models.

2.3.2 End to end learning approaches

The most common approach to skin lesion classification is through transfer learning. However, end to end learning can make sense in specific contexts. For instance, when data and computational resources is not scarce.

In 2019, Ly et al. [17], trained multiple models from scratch with the intention of deploying such models for offline usage in smartphones. They justified this decision by arguing that using pre trained models with large neural network architectures requires a lot more parameters than models trained from scratch. Their best model attained 86% accuracy, significantly better than other transfer learning approaches, while being much more compact (29M). However, they used a huge dataset titled ”PHDB” which was composed of multiple other datasets and contained 80,192 labeled images, which explains the high performance.

3 Methods and materials

The following sections are structured as follows. Section 3.1 provides an explanatory view into the most used image recognition neural network typology. Section 3.2 describes two of the most popular CNN architectures. Section 3.3 explores the concept of repurposing pre-trained models from CNN architectures trained on generic datasets. Section 3.4 explores problems related to training deep networks and some solutions. Finally, Section 3.5 explores state of the art frameworks for training and deploying deep networks.

3.1 Convolutional neural networks

Artificial Neural Networks (ANNs) compose a category of machine learning algorithms that are inspired by biological neural networks. These structures are composed by multiple layers, each composed by multiple neurons that can be interpreted as a function described by parameters. We can look at ANNs as an iterative process that tries to optimize parameters in order to minimize a cost function.

ANNs can approximate any function with few layers, but for more complex problems it deviates a lot from true predictions. Therefore, networks with many more layers and with an organization which allows them to create levels of abstraction are often used to solve more complex problems. We call such networks deep neural networks.

There are many different typologies of ANNs. The most common has a fully connected structure between layers, where each neuron is connected to every other neuron in the previous layer. However, in image recognition such network architecture does not take into account the spatial structure of images. For example, it treats input pixels which are far apart and close together on exactly the same footing.

Instead, convolutional neural networks (CNN) or some close variant are used in most neural networks for image recognition problems [18]. They still retain the core concepts of ANNs, but add 3 different concepts which distinguish them from conventional ANNs:

- Local receptive fields: Each neuron in the first hidden layer will be connected only to a small region of the input neurons.
- Shared weights: Weights and biases are shared across the hidden neurons so that convolutional networks become well adapted to translation variances in images. The shared weights are often said to define a kernel or filter, while to the map from the input layer to the hidden layer we call feature map, where a feature detected by a hidden neuron is some kind of input pattern that will cause the neuron to activate. To do image recognition we need multiple feature maps in order to recognize multiple features.
- Pooling layers: These layers simplify the information in the output from the convolutional layer by removing unnecessary information, such as noise. A common pool layer is max pooling which provides a way to know if a given feature is found anywhere in a region of a image [18].

3.2 Convolutional neural network architectures

Over the years several CNN architectures have been developed and tested against state of the art benchmark challenges such as the Imagenet Large Scale Visual Recognition Challenge (ILSVRC) [19]. In 2012, Krizhevsky et al. [20] submited for the first time a CNN architecture (AlexNet) which outperformed hand-crafted feature learning on the ImageNet. It contained 8 neural network layers, 5 convolutional and 3 fully-connected. This laid the foundation for traditional CNNs: a convolutional layer followed by an activation function followed by a max pooling operation.

Following AlexNet main ideas, the VGGNet [21] was created and became quite popular by winning the 2014's ILSVR. This architecture proved that representation depth is beneficial for the classification accuracy, by using the traditional convolutional network architecture but with increased depth along with smaller receptive fields. There are some public variations of this network, one of which having 16 weight layers (VGG16) displayed in Fig. 1. This architecture is composed by multiple sets of convolutional layers followed by pooling layers that build progressively more abstract features, and at the end a fully connected structure to convert the results of the convolution into a label.

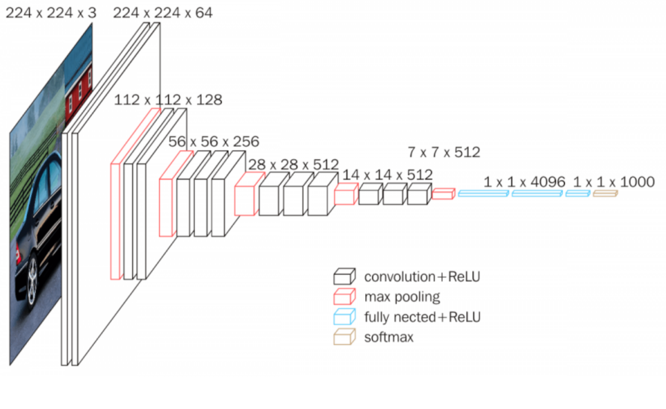

Fig. 1. Architecture of the VGG16 convolutional neural network [21]

3.3 Transfer learning

Supervised learning using deep neural networks requires large amounts of data and computational power in order to train models. However, both of which can either be impossible to have or quite difficult to acquire for small teams. However, even when one has a good dataset along with high computational power, the training process can take a long time, especially while debugging the network to determine a good model fit.

As such, transfer learning techniques are usually used to solve image classification problems [17], which is a common concept used to minimize the effects of the aforementioned problems. Transfer learning is a method of reusing a pre trained model's knowledge for another related task [22]. In deep learning, using transfer learning means to carry the parameters from pre-trained models such as the architectures presented in Section 3.2 trained on a generic datasets such as ImageNet and using those to train another model with a different purpose.

In CNNs, as inputs are passed along the network hidden layers closer to the input layer output generic features like shapes and curves, while hidden layers closer to the output layers build more abstract features such as a dog's face. In order to adapt the pre-trained models into a different domain, one must extract the parameters up to some layer from the pre-trained model while freezing (to not allow parameter updates while training) some or no portion of those layers. As layers near the input layer output generic features, their parameters are usually extracted and potentially frozen, while hidden layers near the output layer are usually not extracted and not frozen, because they output more abstract problem specific features. Additionally, one can expand the original pre-trained model architecture with their own classifier on top, in order to adapt the model into the specific domain.

3.4 Overfitting and underfitting

While training, one must fine tune the model to both accurately make predictions from the training data while generalizing to new data. The bias and variance trade off is a well known problem in deep learning that represents a trade off between these two requirements. While the bias of a model is the error caused by the assumptions made to approximate the model to the true predictions, the variance of a model is the error from sensitivity to small fluctuations in the training set. We must find a good trade off between bias and variance so that the model doesn't underfit or overfit.

If the model underfits then it does not perform well even on the training data, and therefore has high bias and low variance. However, a common problem is to produce a model that performs well on the training data but that generalizes poorly to new data [23]. In this case, we say that the model overfits and therefore has low bias but very high variance. In order to evaluate whether a model is underfitting or overfitting one should use state of the art metrics which help describe what is happening while training.

Multiple solutions to the overfitting problem have been proposed and tested over the years. One common way of dealing with this problem are the regularization techniques, which are broadly described by some authors as any technique that allows the model to generalize better. For example, L1 and L2 regularization attempt to create less complex models [24], while techniques such as dropout ”reduce complex co-adaptations between neurons” [25]. Other methods such as data augmentation can also minimize this problem.

3.4.1 Expanding the training data

In deep learning, a model is highly dependent on its training dataset in order to achieve good performance. A bad dataset can easily cause the network to overfit because it does not provide enough proper real world examples for the network to produce a good bias variance trade off. A good dataset has to represent the real world it tries to describe, be diverse, and most importantly, it needs to have a good number of examples. There are several datasets available which are labelled for skin lesion diagnosis, but a lot of them are quite biased towards some specific class or lack large amounts of examples for a specific class. As such, when some real world variation is introduced the network fails to predict the class.

One way to improve the training dataset with low costs is through a concept called data augmentation. The main idea behind this concept is to expand the training data by applying operations that reflect real-world variation [18], which in turn introduces diversification and size to the dataset. The simpler approach is to apply general transformations, such as translations, rotations or flips to existing samples to create new ones. Another more complex approach is to synthetically create new images based on some original dataset (generative models) through methods such as generative adversarial networks, a type of neural networks.

3.5 Deploying deep learning models

Deploying deep learning models requires careful orchestration of components such as a learner for generating models, a visualization tool to analyse and validate models and a serving framework which exposes models through an API. Usually, these components are hard coded together by custom scripts leading to high coupling and low cohesion. This poses problems for future expandability of such systems because simple changes can completely break the pipeline. Therefore, it is a priority to build these systems in a modular approach such that components are independent of each other. In addition, requirements such as easy-to-use configuration and tools, scalability and reliability should also play a big role when considering frameworks and tools to create a cohesive architecture for a production application.

Frameworks such as Tensorflow Extended [26] attempt to integrate the aforementioned components and requirements into one platform and standardize the whole process. Tensorflow has become a more production centered platform over the years, for example, by integrating Keras [27] into it, which provides more easy to use high level concepts to train and test models. There are other options such as Theano [9] or pyTorch [28] but these are more focused around research environments.

Training deep learning models usually requires high computational requirements, which is why nowadays most of these frameworks take advantage of GPUs through the CUDA platform. However, for small teams such computational power might be inaccessible or the cost of either time or money to setup such system might be too much. In such cases, it is better to take advantage of cloud services to train these models.

4 Proposed Work

The proposed work focuses on two main tasks:

- Train and test a multi class deep learning model for skin lesion classification that empowers early intervention over skin cancer. Such task will require the studying of different pre-trained models, as well as hyperparameter and model optimization. Data augmentation can also play a big role on the model's performance therefore should be carefully studied.
- Develop a responsive eHealth app for patients and dermatologists that integrates two components which can be seen in Fig. 2. The first is a black box change detection tool for patients that notifies users whenever something is wrong in their skin and establishes a communication channel with dermatologists. The second is a decision support tool for dermatologists on clinical environments that contains the aforementioned classifier of skin lesions.

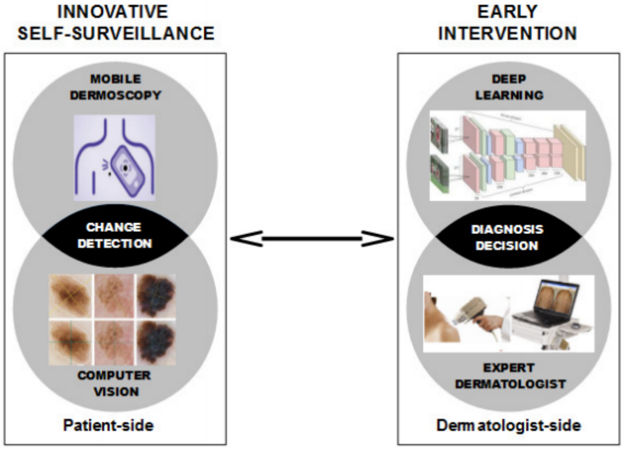

Fig. 2. Proposed eHealth application diagram

5 Conclusion

As the incidence of skin cancer rises, there is a clear need for skin lesion diagnosis tools integrated within eHealth applications that provide support for patients and health professionals. At the same time, new advancements on deep learning methods allow near dermatologist performance with high margin for improvement which overshadow other methods. Challenges such as the requirement for large datasets or the high computational requirements hamper the performance of models and need to be addressed before deploying such tools into production. However, promising techniques such as transfer learning and data augmentation prove to minimize the effects of such factors. Finally, it is expected that these issues will become less relevant as more labeled skin lesion data becomes publicly available.

References

[1] "Skin Cancer Facts & Statistics - The Skin Cancer Foundation," 2019. [Online]. Available: https://skincancer.org/skin-cancer-information/skin-cancer-facts/

[2] F. Bray, J. Ferlay *et al.*, "Global cancer statistics 2018: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries," *CA: A Cancer Journal for Clinicians*, vol. 68, no. 6, pp. 394–424, nov 2018.

[3] A. Esteva, B. Kuprel *et al.*, "Dermatologist-level classification of skin cancer with deep neural networks," *Nature*, vol. 542, no. 7639, pp. 115–118, feb 2017. [Online]. Available: http://www.nature.com/articles/nature21056

[4] H. A. Haenssle, C. Fink *et al.*, "Man against Machine: Diagnostic performance of a deep learning convolutional neural network for dermoscopic melanoma recognition in comparison to 58 dermatologists," pp. 1836–1842, 2018.

[5] J. Jaworek-Korjakowska and P. Kleczek, "ESkin: Study on the smartphone application for early detection of malignant melanoma," *Wireless Communications and Mobile Computing*, vol. 2018, 2018.

[6] "DermEngine — Visual Search." [Online]. Available: https://www.dermengine.com/en-ca/visual-search

[7] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

[8] H. Greenspan, B. Van Ginneken, and R. M. Summers, "Guest Editorial Deep Learning in Medical Imaging: Overview and Future Promise of an Exciting New Technique," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1153–1159, may 2016.

[9] F. Bastien, N. Pascal Lamblin *et al.*, "Theano: new features and speed improvements."

[10] J. Deng, W. Dong *et al.*, "ImageNet: A large-scale hierarchical image database." Institute of Electrical and Electronics Engineers (IEEE), mar 2010, pp. 248–255.

[11] T. J. Brinker, A. Hekler *et al.*, "Skin Cancer Classification Using Convolutional Neural Networks: Systematic Review." *Journal of medical Internet research*, vol. 20, no. 10, p. e11936, oct 2018. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/30333097

[12] P. Tschandl, C. Rosendahl, and H. Kittler, "The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions," *Scientific Data*, vol. 5, aug 2018.

[13] "ISIC 2019." [Online]. Available: https://challenge2019.isic-archive.com/

[14] "Task 3: Lesion Diagnosis — ISIC 2018." [Online]. Available: https://challenge2018.isic-archive.com/task3/

[15] A. Nozdryn-Plotnicki, J. Yap, and W. Yolland, "Ensembling Convolutional Neural Networks for Skin Cancer Classification."

[16] N. Gessert, M. Nielsen *et al.*, "Skin Lesion Classification Using Loss Balancing and Ensembles of Multi-Resolution EfficientNets."

[17] P. Ly, D. Bein, and A. Verma, "New Compact Deep Learning Model for Skin Cancer Recognition," aug 2019.

[18] M. Nielsen, *Neural Networks and Deep Learning*, 2018. [Online]. Available: http://neuralnetworksanddeeplearning.com/

[19] O. Russakovsky, J. Deng *et al.*, "ImageNet Large Scale Visual Recognition Challenge," pp. 211–252, dec 2015.

[20] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, jun 2017.

[21] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks For Large-Scale Image Recogntinion," 2015. [Online]. Available: http://www.robots.ox.ac.uk/

[22] T. G. Dipanjan Sarkar, Raghav Bali, *Hands-On Transfer Learning with Python*, 1st ed. Packt Publishing, 2018.

[23] J. Grus, *Data Science From Scratch*. O'Reilly Media, 2015.

[24] A. Ng, "Feature selection, L 1 vs. L 2 regularization, and rotational invariance."

[25] G. E. Hinton, N. Srivastava *et al.*, "Improving neural networks by preventing co-adaptation of feature detectors," jul 2012. [Online]. Available: http://arxiv.org/abs/1207.0580

[26] D. Baylor, E. Breck *et al.*, "TFX: A TensorFlow-Based Production-Scale Machine Learning Platform," 2017. [Online]. Available: http://dx.doi.org/10.1145/3097983.3098021

[27] F. Chollet and Others, "Keras," \url{https://github.com/fchollet/keras}, 2015.

[28] A. Paszke, S. Gross *et al.*, "Automatic differentiation in PyTorch," Tech. Rep., 2017.