First Year Project 2

Lesion Detection

Alexander Nielsen
alwn@itu.dk

Christian Hetling
chrhe@itu.dk

Erling Amundsen
erla@itu.dk

Krzysztof Parocki
krpa@itu.dk

Malthe Musaeus
mhmu@itu.dk

IT UNIVERSITY OF CPH

April 8, 2022

Contents

List of Figures

List of Tables

1 Introduction

A timely diagnosis of skin cancers, like melanoma, is one of the most important factors in dermatology. For most patients, early diagnosis of skin cancer means a safe recovery. However, dermatology is one of the most challenging fields in terms of diagnosis; even trained dermatologists often need to review patient history and conduct further tests in order to make a proper diagnosis. Therefore, a faster method of classifying skin diseases, such as melanoma, is essential. Machine learning (ML) and artificial intelligence (AI) have seen rapid growth in use cases over the last decades. The set of applications where ML is especially effective is classification problems, as an ML algorithm can be used to categorize objects based on their measurable features. Medical imaging is an example of a field where ML has been implemented to diagnose patients effectively using classification.

The purpose of this study is to extract features from the ISIC 2017 dataset and train a classification model based on them. Following the ABC (Asymmetry, Border, Color) of skin cancer, we chose the asymmetry and color variance of patients' lesions as the main characteristics of interest. After extracting these features from the pictures, the model should be able to predict how likely a new patient is to have melanoma (officially malignant melanoma) based on their skin lesion.

In this paper, we will present the process of extracting our own features from the ISIC images and segmentation masks, as well as the results of training machine learning models on the extracted features (with both the K Nearest Neighbours (KNN) and the Decision Tree (DT) algorithms) and using them to predict if unseen lesions are characteristic of melanoma or not. Finally, we will compare the results of our KNN predictions to the results given by a Convolutional Neural Network (CNN).

2 Data

The data used to train our model was taken from the ISIC 2017 data set. The base (truncated) dataset consists of 150 dermoscopic images of skin lesions with an associated set of segmentation masks. The segmentation masks are used to distinguish the area of the lesion from the skin. The data also includes a training_ground_truth file, which is a single CSV file containing 3 columns: image ID, binary classification of melanoma, and binary classification of seborrheic keratosis. The 150 ISIC pictures are a part of a larger dataset, which contains 2000 pictures, corresponding segmentation masks, and a ground truth file.

	Base set (150)		Full set (2000)	
Label	Count	%	Count	%
Nevus	78	52.0	1372	68.6
Melanoma	30	20.0	374	18.7
Seborrheic Keratosis	42	28.0	254	12.7

Table 1: Distribution of labels in the ISIC datasets.

The distribution of labels in the truncated dataset (first column) and in the full dataset (second column) can be seen in Table 1. Melanoma makes up about 19-20% of both datasets. A challenge we can instantly see is the inclusion of seborrheic keratosis, which is a non-cancerous skin lesion that shares some visual features with Melanoma, the main similarites being size and color. On the other hand, our models make no attempt at diagnosing seborrheic keratosis and our feature extraction focuses on symmetry and color variance, which are not neccessarily shared between melonoma and seborrheic keratosis.

Before writing the implementations, we went through the data and labeled features of 50 images manually. We judged Assymetry, Border, Color (from Abcde's of melonoma) binarily as melanoma or not melanoma.

For some of the lesions all 5 of us agreed, and on some there was a bit of debate on what to rate them. Through the manual process it is hard to always discern different ratings from each other the same way. Therefore automatic feature extraction seem like a more consistent alternative.

3 Methods

As the ISIC pictures ranged in size and quality, we scaled them down to the size of the smallest image in the dataset (720p HD) before extracting any features. Additionally, the masks were often inaccurate in distinguishing the mole from the skin, so we constructed our own method to create more accurate masks, and used them predominantly to compute color variance.

Our own masking function takes the dominant colour found in a picture and assumes that to be the skin colour. It then removes all colors similar to the skin colour inside the original segmentation. This way, we can get a more accurate color variance of the mole. The masking function has some limitations in finding the proper mask if the dominant colour of the picture is the colour of the mole. It then removes every pixel resembling the colour of the actual mole - this drawback is further explored later in the paper.

3.1 Feature Extraction

To choose which features to base our classification on, we used the ”ABCDE of melanoma” from the American Academy of dermatology.[1] This led us to decide to extract three main features: circularity, asymmetry and color variance.

3.1.1 Circularity

The circularity of the lesion was calculated using the original binary masks, which had been created by clinicians using either manual or semi manual methods [2]. First, the area and perimeter of a skin lesion were calculated. Area was calculated as the total pixel area of a given mask, and to measure the perimeter we subtracted an ”eroded” binary mask (a mask shrunk by one pixel on each side) from the original mask to get a lot of single pixels that could be summed, returning a perimeter value. Then, we used a formula for circularity $\frac{4*\pi*\text{Area}}{\text{Perimeter}^2}$, which is 1 for a perfect circle and goes down towards 0 for highly non-circular shapes. As the formula returns a ratio, we can assume that the result is standardized (it’s not dependent on the total number of pixels, i.e., the resolution of the pictures).

3.1.2 Symmetry

To extract symmetry, we used the original segmentation masks provided in the dataset. For each lesion, the mask was split in two in its center of mass. Then we overlapped the two sub-masks to see how well they matched each other. This process was repeated 180 times for 180 degrees, returning two values as percentages of the original mask area - the minimum value of the non-overlapping areas and the average value of the non-overlapping areas. Running the symmetry function on a perfectly symmetrical circle returns a minimum score of 0.19% and an average score of 0.39%, whereas running it on an equilateral triangle, a shape symmetric on three axes, returns a minimum score of 0.67% and an average score of 23.6%. After running the symmetry function on 2000 pictures, excluding pictures where the mask extended outside the picture, the highest minimum symmetry score was 28.8% and the highest average symmetry score was 56.3%.

By returning percentage values, we achieve standardization that is unaffected by the overall quality of the picture, as well as the distance the picture was taken from. Additionally, for segmentation masks that

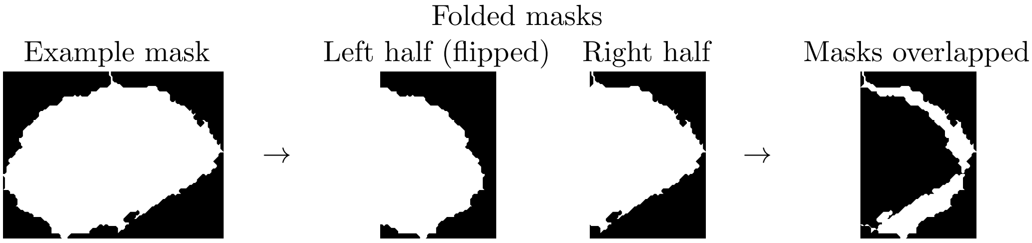

Figure 1: Symmetry function visualized

extend out of the picture it would be impossible to measure the symmetry precisely. Those masks were automatically given a maximum possible score of 100% in both the minimum and average scores.

3.1.3 Color Variance

Another feature we've extracted was color variance of each lesion. We considered two approaches: measuring the number of different colors the lesion consist of, and the biggest difference between two colors found in the lesion. After a lot of testing, we decided to use the second approach. If the lesion consisted only of two distant colors (like black and light red), the color variance would obviously be huge, whereas the first approach would yield a very low result. We used our own masks to separate the lesion from the skin, as they produced better results than the original ones.

To measure the biggest distance between colors, we had to remember that the RGB color space is not perceptually uniform (two very different colors can have values that are relatively close to each other, and vice versa). Therefore, we had to convert the color space of each photo to CIELAB, which is perceptually uniform (a given numerical change corresponds to a similar perceived change in color). Then, to reduce the number of colors to work with, we performed MiniBatch K-Means clustering, effectively reducing the total number of distinct colors to 16. We did this for speed and memory purposes. The number 16 was chosen as optimal value, allowing for a significant increase in the algorithm's speed while still retaining enough color information to measure the differences.

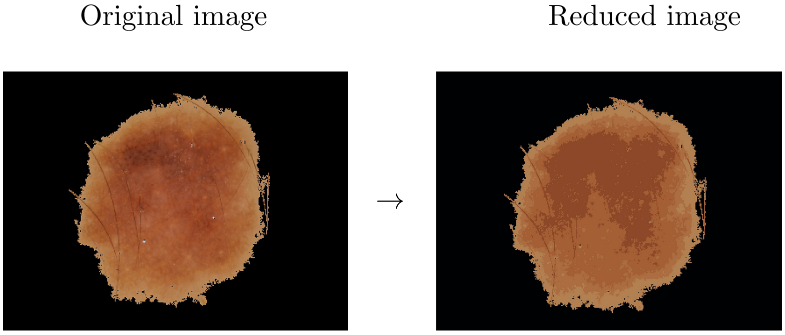

Figure 2: Color reduction visualized

Finally, we measured distances between the colors still present in the lesion (using the Pytagoras formula for 3 variables), and returned the maximum of all measurements as color variance. The resulting values range from 0 (when the lesion is very color-uniform) to around 200 (when the lesion consists of at least two very different colors). It makes sense given the maximum distance in the CIELAB color space is around 374 units.

3.2 Classification

There are multiple possible models that can be used to predict classifications of data. We chose to classify the skin lesions using the K nearest neighbor (KNN) algorithm.

3.2.1 K Nearest Neighbours

KNN classification works by transforming the extracted features of an image into a feature space. An unseen image is then classified by the most common class among its k nearest neighbors.

As there were significant differences in the values the features we measured were taking on (the value range), we had to achieve some sort of standardization. Otherwise, the sole range of possible values could greatly influence the distance of classes, making one feature more important than the other. Therefore, we scaled every feature using SKlearn's standard scaler from their preprocessing module. By using the standard scaler, we make sure that the means of all features are 0 with a standard deviation of 1.

We ran the KNN algorithm several times with different values of k in order to maximize the efficiency of the model. We found that the model's accuracy peaked at 5 neighbors. When considering more neighbors, there was a higher chance of predicting a benign mole. This could be explained by the skewed $\frac{\text{melanoma lesions}}{\text{benign skin lesions}}$ ratio (in the direction of benign lesions) in the data as seen in Table 1.

3.2.2 Decision Trees

Decision Trees (DTs) are a non-parametric machine learning model which can also be used for classification. The model is often structured as a binary tree. DTs are trained to predict the value of a target variable by learning simple decision rules from the data's features. Each non-leaf node narrows down the possibilities until you arrive at a leaf-node, which contains a predicted label.

3.2.3 Neural Network

Both KNN and DTs are based on features that have to be extracted from the data beforehand. Another type of classifier is a neural network. In the case of image classification, a convolutional neural network (CNN) specifically would be an ideal choice. CNNs work by having a matrix of weights that convolve across an array of pixels color values. The training part of the model then figures out the weights to use to extract features from the image. This is done by calculating new weights after each iteration using gradient descent of a chosen loss metric, like for example binary cross entropy. CNNs do not require features to be extracted before training or predicting, in contrast to the KNN and DT models. Rather a CNN expects an image, in the form of an array, as input and outputs a given target variable.

4 Evaluation

When training any model it is important to evaluate the results of the predictions afterwards. The final evaluation was done on unseen data to avoid over-fitting. There are a lot of different possible metrics that can be used to approximate the efficiency/accuracy of a model's prediction. Therefore, it is essential to pick a metric that is relevant for the specific use case of the given model.

4.1 Accuracy vs. ROC

While evaluating the models we looked at three main measurements of prediction accuracy for both our models: accuracy, The Receiver Operating Characteristics Area Under Curve (ROC AUC) and the F1 score.

The prediction accuracy from the KNN algorithm returned a score of .805, and the decision tree returned a score of .712, which is calculated by taking number of correct guesses over the amount of guesses. In the case of the ISIC datasets, the accuracy score is not a good predictor, as the percentage of pictures showing a melanoma lesion is 18.7% (for the 2000 images). As seen in the KNN confusion plot in Figure 3, the KNN achieves a high accuracy due to the model mainly predicting lesions as non melonoma, indicating that the KNN model's fails in regards to accurately diagnosing cases of melanoma. In regard to accuracy, the CNN model was outperformed by the decision tree, scoring .75%.

The Receiver Operating Characteristics (ROC) is a better measurement of the predictions, as it measures the degree of which the algorithm manages to separate two classes (in this case, melonoma and non-melonoma). The ROC is a curve comparing true positives (predicted positives being correct), and false positives (predicted positives being incorrect). The value measured is the area under the curve (AUC) of the ROC. An AUC score of 0.5 means that the algorithm is unable to separate the two classes, and scores between 0.5 and 1 shows the degree of which the algorithm manages to separate the classes and label them correctly.

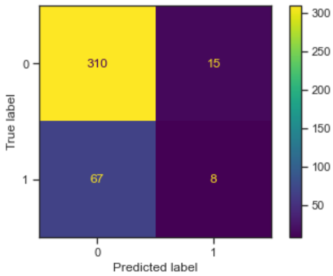

Figure 3: KNN confusion plot

The KNN algorithm returned an AUC score of .530, and the Decision tree returned a score of .508. These numbers reflect that both the KNN and the decision tree algorithms struggled to accurately separate the the lesions into the non-melanoma and melanoma groups. More precisely, the KNN algorithm would diagnose a random picture of a melanoma correctly 53.0% of the time, and the Decision tree 50.8% of the time. In this regard, CNN (Convolutional Neural Network) heavily outperforms both KNN and the decision tree, scoring .7689.

The F1 score is calculated as the harmonic mean of precision and recall, precision being the number of true positives over all positive guesses, and recall being the amount of true positives over the amount of all positive pictures. The F1 score is similar to the accuracy score, however, it is more sensitive to false negatives and it is a better fit for our dataset, as it is not skewed by the imbalance of pictures with and without cancer.

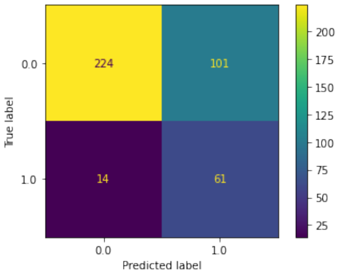

Figure 4: CNN confusion plot

Additionally, in our case of predicting cancer, occurrences of false negatives are dangerous, so it's advisable to use a metric taking them into account. The F1 score requires both good predictions of true positives and a low amount of false negatives. The F1 score for the KNN is .163, and .177 for the decision tree. After comparing these scores to that of the CNN (being .480), it is evident that the CNN outperforms in regards to correctly labeling a lesion as seen on Figure 3 and Figure 4.

4.2 Cross-validation

To avoid overfitting of our classification model, we used a stratified K-Fold Cross Validation(sKCV). SKCV ensures that there is an equal amount of both cases (melanoma and non-melanoma) in the validation sets and training sets, and reduces the chance of overfitting. Seeing that sKCV splits the cases equally, we found it to be the best fitting for our dataset (in which only 18.7% of cases were melonoma), and we found that we got the best results when using an sKCV value of 6.

4.3 K Nearest Neighbors vs. Convolutional Neural Network

There are some clear differences between the models we trained with KNN and CNN, respectively. The distinction can be seen both in regards to the effectiveness of predictions, but also in the amount of preprocessing required for each of the models. The KNN and DT models both required us to extract features from the images and then pass them to the model. This puts a lot of emphasis on extraction of proper and usable features, as well as requires some level of knowledge on the explored domain of science. In contrast, CNN processes the images as arrays of pixels and learns the patterns of data on its own.

After training both types of models, we have come to the conclusion that CNN is a quick way to get a decent model without knowing that much about the domain. In some sense, it's a blackbox that learns the patterns without external help. On the other hand, with domain knowledge, it would be possible to extract usable features, and use them to train a model which ”understands” the subject better. Both are viable options, and the choice should depend on the knowledge about the subject.

5 Limitations

We've identified two main limitations of our project. First and foremost, it's important to acknowledge the fact that to properly asses features of the lesion, one needs to posses some amount of domain knowledge. Our knowledge of dermatology wasn't big, and therefore we might have interpreted something in a wrong while extracting features. To counter this, we decided to compare our results to a CNN model. However, neural network is not a panacea for solving the issue of feature extraction.

Another limitation we identified was related to our own masking algorithm. The provided segmentation masks often failed in distinguishing the skin from the lesion accurately, resulting in a much higher color variance of the lesion in certain pictures (due to the skin color being taken into consideration). Therefore, we used our own function to compute segmentation masks. However, our function is based heavily on color differences, resulting in some pictures being masked wrong (this happens in some of the significant close-ups, where the most prominent color in the image is the color of the lesion, not the color of the skin). On the other hand, the number of times this happened was very low, so we decided to still use our function to get better color variance scores for most pictures. For the symmetry check, the original masks were passed as the input, as we felt the original segmentation did a good job at representing the shape of the lesions.

6 Conclusions

As addressed in the evaluation, both the KNN and Decision Tree models struggle to separate the two classes of lesions into their own groups. The scores (AUC in particular) for both models reflect that the models were able to distinguish some cases, however, not to the extent of accurately diagnosing melanoma. There were likely some faults in the feature extraction step, and the extracted features were insufficient for training reliable ML models. To compensate for that, we showed that a CNN trained on segmented skin lesion pictures could reach high scores (in particular, an AUC of .7689) despite lacking the supervision present in the KNN model. However, a CNN model is not a panacea - someone's life could depend on it making the correct prediction, and with the F1 score of .480 it wouldn't be reasonable to trust the algorithm.

7 Disclosure

When loading pictures for our feature extraction it is important that the masks are loaded trough the Open Cv library, ”cv2.imread(imgpath)”, and that the masks are read in gray scale, ”cv2.imread(imgpath,0)”.

References

[1] Abcde´s of melonoma. https://www.beaumont.org/conditions/melanoma/abcde's-of-melanoma: :text=ABCDE%20stands%20for%20asymmetry%2C%20border,the%20shape%20isn't%20uniform.

[2] Codella, Gutman, Celebi, Helba, Marchetti, Dusza, Kalloo, Liopyris, Mishra, Kittler, and et al. Isic challange datasets. *Skin Lesion Analysis Toward Melanoma Detection: A Challenge at the 2017 International Symposium on Biomedical Imaging (ISBI), Hosted by the International Skin Imaging Collaboration (ISIC)*, 2017.