# Detecting Melanoma from images of moles/spots using Convolutional Neural Networks and SVMs

**Jason Hallman, Vikram P.T**

**University Of North Carolina at Chapel Hill**
**COMP 562, Final Project**

## ABSTRACT

Melanoma is a serious form of skin cancer that begins in cells known as melanocytes. Even though it may not be as prevalent as basal cell carcinoma (BCC), it could still be deadly as it can spread to other organs of the body very rapidly if not diagnosed and treated at an early stage. Therefore, we decided to help address this problem using Convolutional neural networks (CNN) and Support Vector Machines (SVM) to classify whether a given image of a spot of melanoma is benign or malignant. We also compare each of our models and discuss them in detail in this paper.

**Index Terms**- Convolutional Neural Network (CNN), Melanoma, Support Vector Machine (SVM), ROC, Machine Learning (ML).

## PROBLEM & GOAL

Melanoma begins at the melanocytes. Melanocytes are cells found in the upper layer of the skin. They also produce the pigment Melanin that gives color to your skin. The exact cause of all melanoma is not clear, but we know that exposure to ultraviolet (UV) radiation from sunlight or tanning beds causes mutation in the DNA of the melanocytes and this results in uncontrolled cellular growth. The American Cancer Society predicts that there will be 100,350 new patients diagnosed with melanoma in 2020 and about 6850 people are expected to die because of it. Therefore, we hope to apply some machine learning techniques to aid us in diagnosing melanoma in its early stages so it can be treated efficiently.

## DATA

We found the data on Kaggle's website. This data belongs to the ISIC archive and is bound to the ISIC archive rights and thus our work on this data is not intended to be monetized.

The data came pre-split as training and testing sets. The training set consisted 2637 images of malignant and benign moles. The testing set consisted 660 images of the two types of moles. Both the train and test data were balanced in terms of images of benign and malignant moles. Each of these images are colored RGB images of dimensions 224x224x3.

### WHAT MAKES A MOLE BENIGN OR MALIGNANT?

Before applying an ML algorithm we viewed a few examples of malignant and benign moles to see exactly what made them different. In general, malignant moles are more uneven, asymmetrical and have coarser texture. However, there are exceptions. Malignant moles vary greatly and may have all, one or none of the characteristics mentioned above. It is our task to engineer features from the images

and make some useful and accurate predictions.

## MOTIVATION AND RELATED WORK

We understood that to use an SVM as a classifier we would have to use some feature engineering methods to extract features from the images we had. The idea behind feature extraction is that feeding characteristic features of the signals—rather than the signals themselves—to a trainable classifier improves performance. We referenced articles and papers about feature engineering methods and decided to use the Gabor and Local binary pattern (LBP) filters which are textural feature extractors.

## MODELS AND METHODS

### 1. SVM

We converted the RGB images to grayscale images (224x224x1) and used python's sklearn package to apply the Gabor and LBP filters. The Gabor transformation filter is a linear filter used for texture analysis. It analyzes whether there are any specific frequency content in the image in specific directions in a localized region. The LBP is an effective texture descriptor for images which thresholds the neighboring pixels based on the value of the current pixel. LBP descriptors efficiently capture the local spatial patterns and the gray scale contrast in an image. Each of these filters output an image matrix of the same dimension (224x224x3). After viewing these images, you understand exactly what these filters are trying to do.

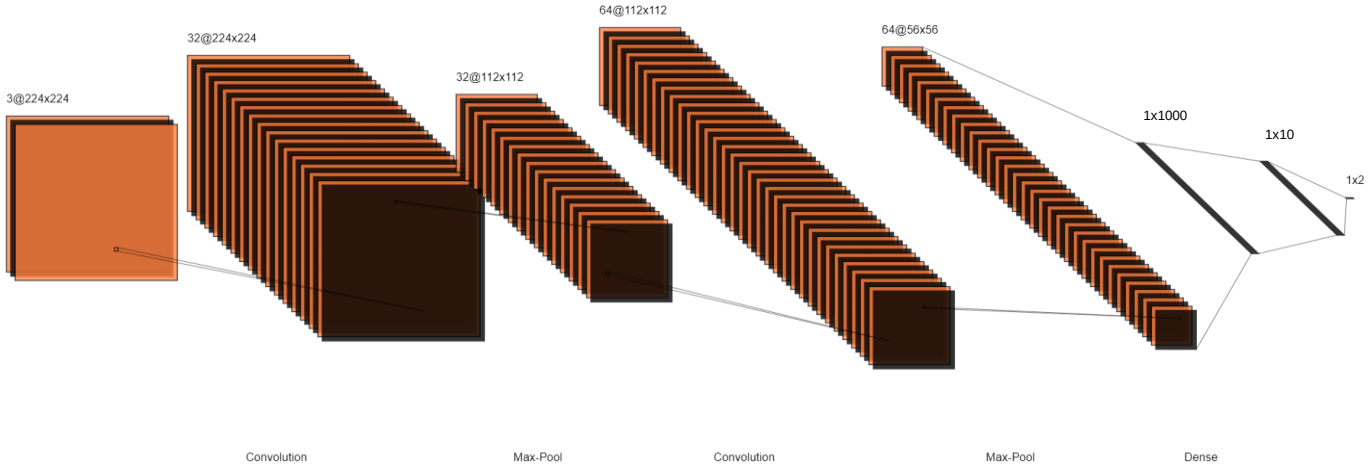
We summarized the output images using histograms (counts of the number of pixels with values in 12 equal ranges between 0 and 255). We used the histograms as our feature vectors for the images. Therefore each image now had 12 x 2 = 24 individual features. For details about how the image looked after applying the filters, please look at our GitHub pages as the images might be a bit unsettling for some people. We used sklearn's SVM classifier to train our model. The SVM classifier seemed to perform best with a polynomial kernel, so we decided to use that for our analysis.

### 2. CNN

CNNs do all of the feature extraction automatically and are more efficient than these hand engineered methods.

In addition, because of the reasonably small size of the images, there was no need to resize them for training. We chose to take two separate approaches to training a CNN for malignancy classification. First, we used an architecture with a small number of nodes and without any pre-trained weights. Second, we took a pre-trained ResNet50 and retrained the classification layer on our data. We used Pytorch to build and train both models which was done in Kaggel's GPU accelerated cloud notebook. In order to keep results comparable across models we fed the raw, 224x224x3 RGB images into both.

Both models used the same architecture for their classification layer. On both models we used Cross Entropy Loss for the loss function and ADAM as our optimization algorithm. The architecture for our CNN consisted of two sequential layers each of which had a Conv2d, ReLU, and MaxPool2d. The first convolutional layer took the image from 3 channels to 32, and the second took it from 32 to 64. Both used a kernel size of 5, a stride of 1, and a padding of 2. Both max pools used a kernel size of 2 and a stride of 2. For classification, we used 3 fully connected layers which took an input of 64 x 56 x 56, reduced it to 1000, and reduced it again to 10, and finally outputted to 2 nodes for classification. Both were trained for 3 epochs with a relatively small batch size of 32 to prevent exhausting the memory on our GPU.




## RESULTS

### 1. SVM

Our model achieved a 69.85% accuracy rate on our test data. The confusion matrix is shown below.

| | **Malignant** | **Benign** |
|---|---|---|
| **Malignant** | 167 | 133 |
| **Benign** | 66 | 294 |

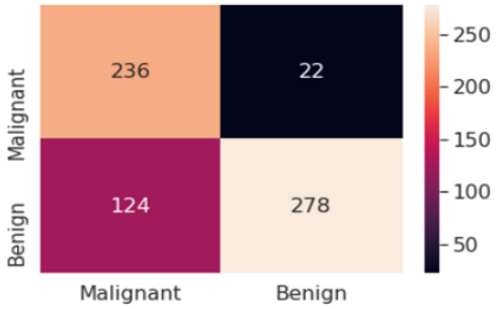
Sensitivity = $\frac{167}{167+133}$ = 55.67%

Specificity = $\frac{294}{66+294}$ = 81.67%

False-Negative rate= $\frac{133}{133+167}$ = 44.33%The results tell us that our model is not great at classifying moles that are malignant (55.67% accuracy) but it is relatively much better at identifying the moles that are benign (81.67% accuracy). Furthermore, we have a high false negative rate which is not desired in a medical models such as this. These results verify the fact that malignant spots/moles are much harder to detect and are of varying characteristics. Therefore, we are going to need to build a more advanced model. We look to CNNs to minimize the false-negative rates and improve overall accuracy.

### 2. Our CNN architecture

Our model achieved a 77.88% accuracy rate on our test data. The confusion matrix is shown below.



Sensitivity = $\frac{236}{236+22}$ = 91.47%

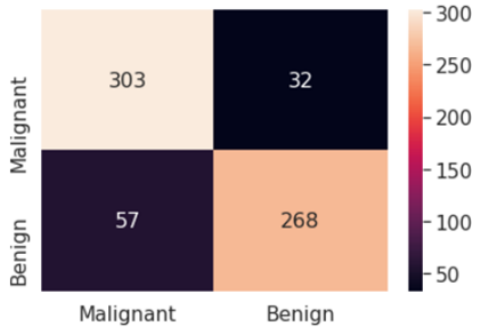
Specificity = $\frac{278}{124+278}$ = 69.15%

False-Negative rate= $\frac{22}{236+22}$ = 8.53%

Our CNN architecture shows a marked improvement in accuracy (77.88% vs 69.85%) and sensitivity (91.47% vs 55.67%) when compared to the SVM. As well, the false-negative rate was reduced greatly from the SVM (44.33%) to the CNN (8.53%). This is an important improvement for its practical use as a diagnostic model. One area in which the SVM (81.67%) outperforms our CNN (69.15%) is in specificity showing that the SVM is better at correctly classifying benign moles.

### 3. ResNet50

Our model achieved an 86.52% accuracy rate on our test data. The confusion matrix is shown below.




Sensitivity = $\frac{303}{303+32}$ = 90.45%

Specificity = $\frac{268}{57+268}$ = 82.46%

False-Negative rate= $\frac{32}{303+32}$ = 9.55%

The pre-trained ResNet50 model with our classifier on top, shows increased accuracy (86.52%) over the tested accuracy of our CNN (77.88%). The sensitivity between the two models is roughly the same (90.45% vs 91.47%). The ResNet50 model has better specificity (82.46%) than our CNN (69.15%), indicating that the ResNet 50 model is better at classifying benign moles than our CNN. In addition, the misclassification rate for malignant moles was three times that of benign ones in our CNN. When compared to the ResNet50 model which misclassified benign at a rate only twice that of malignant. Overall, we see that the ResNet50 model has errors weighted much more evenly between malignant and benign when compared to our CNN where the misclassification of benign was much higher than that of malignant. This has led to the ResNet50 model having a higher false-negative rate (9.55%) than our CNN (8.53%), but the difference is small and could be cause by the specific images chosen for the test data.

## CONCLUSION

As was mentioned earlier, we chose to evaluate our models by their Specificity and Sensitivity/false negative rates. We observed that the SVM model had low Sensitivity and relatively high Specificity. Our first CNN model had high Sensitivity but low Specificity and the ResNet50 model had high Sensitivity and Specificity which is exactly what we were looking for in our model.

In conclusion, the ResNet50 model performed the best according to our criteria. Additionally, we can see that the CNNs outperform the SVM model with hand generated features. This is a testament to the power of end-to-end learning with Deep CNNs. We believe that the CNNs were more effective because they were able to engineer features tailored specifically to the data which were much more efficient than the general hand engineered feature extraction techniques we implemented. We also feel that if higher resolution images were used, we would have definitely seen an increase in the performance of the models.

## CITATIONS


1. *IITK Deep Learning in Visual Computing NPTEL MOOC pythonlecture notes for feature engineering: https://github.com/iitkliv/dlvcnptel/blob/master/lecture3.ipynb*

2. *Melanoma Overview, April 2019. https://www.skincancer.org/skin-cancer-information/melanoma/*

3. *Melanoma, by Mayoclinc staff. https://www.mayoclinic.org/diseases-conditions/melanoma/symptoms-causes/syc-20374884*

4. *Key statistics for Melanoma Skin cancer: https://www.cancer.org/cancer/melanoma-skin-cancer/about/key-statistics.html*

5. *A mathematical theory of Deep Convolutional Neural Networks for Feature Extraction by Thomas Wiatowski and Helmut Bolcskei https://arxiv.org/pdf/1512.06293.pdf*

6. *Local Binary Pattern: https://www.sciencedirect.com/topics/engineering/local-binary-pattern*

7. *Through the eyes of the Gabor Filter by Anuj Shah: https://medium.com/@anuj_shah/through-the-eyes-of-gabor-filter-17d1fdb3ac97*

8. *Data source:* https://www.kaggle.com/fanconic/skin-cancer-malignant-vs-benign