

Mole Investigator: Detecting Cancerous Skin Moles Through Computer Vision

Dept. of CIS - Senior Design 2014-2015 *

Abhishek Gadiraju
agadi@seas.upenn.edu
Univ. of Pennsylvania
Philadelphia, PA

Sneha Keshwani
snehak@seas.upenn.edu
Univ. of Pennsylvania
Philadelphia, PA

Elise Minkin
minkine@seas.upenn.edu
Univ. of Pennsylvania
Philadelphia, PA

ABSTRACT

Early detection of skin cancer is crucial for patient survival. Often people take no initiative to have moles inspected by doctors because it is time and cost intensive. This shows the need for a mechanism to incentivize at-risk users to improve their chances of survival by seeking medical attention. We filled this need by building an iOS application that allows users to estimate the likelihood that their moles are melanoma by analyzing their mole images and by querying them about familial and environmental risk factors.

The iOS application has an intuitive interface on the mobile frontend and a robust machine learning model on the backend that classifies new images as they are sent in and returns a response to the application in an acceptable amount of time. This machine learning model has an accuracy of 79.8% when classifying images in the test set. The application strives to empower people by allowing them to perform self-tests after they notice a possibly suspicious mole anywhere on their body, as early detection is the key to a high survival rate in melanoma.

1. INTRODUCTION

Skin cancer is an extremely serious and growing problem worldwide. One in five Americans alive today will develop some type of skin cancer during his or her lifetime.[1] Early detection of skin cancer is crucial for patient survival. If melanoma is caught in the "Localized Stage," the five-year survival rate is 98%. However, if melanoma is caught in the "Distant Stage," the five-year survival rate is 15%.[2]

Often, people know they have moles but take no initiative to have them checked by a doctor. 16% of melanoma cases are caught in the distant stage. 76,100 cases of melanoma were diagnosed in 2014. This means 12,176 distant stage melanomas were diagnosed in 2014 and that 12,176 people were told that they have only a 15% chance of living for the next five years.[3] The reason that people do not seek medical attention for their moles could be because it is a hassle to see a doctor and because skin biopsies are invasive and can be expensive.

This shows the need a mechanism to incentivize at-risk people to seek medical attention in order to increase their chances of survival. To fill this need, we built an iOS application that allows users to determine the likelihood that their moles are melanoma by analyzing their mole images and by querying them about environmental risk factors. The mole

images are classified by a random forest machine learning model and this classification is further refined by the questionnaire that is designed to mimic a dermatologist's evaluation. Having an application tell users if moles are likely to be cancerous will give them reason to urgently seek medical help before it is too late.

2. RELATED WORK

2.1 Examinations by Doctors

2.1.1 Standard Processes

At present, the gold standard for skin cancer diagnosis is a visual examination by a dermatologist followed by a biopsy.[4] A biopsy involves analyzing a sample of skin from the area of concern under a microscope to determine with certainty whether the mole is cancerous. Biopsies are invasive, expensive and time consuming to perform. As a result, doctors try to use visual examinations and other techniques to determine whether a biopsy is actually necessary in a given case.

During a visual examination, the doctor will generally take into account the asymmetry, border irregularity, color irregularity, and diameter of the mole. The patient will also be asked for his/her medical history and other relevant questions such as how much sun exposure the patient has had and whether the mole changed over time. In addition, the doctor will inquire about recent exposures to known causes of skin cancer and if the mole has been changing over time. The rest of the patient's body may be examined for any other spots or moles that could be related to cancer.[4]

Our solution includes both image analysis and a questionnaire that encompasses many of the features that dermatologists consider. This allows us to calculate the probability that the mole is malignant and make a recommendation on whether or not the user should seek medical attention.[5]

Our solution attempts to mimic a dermatologist's examination but does not attempt to replace dermatologists and biopsies. Instead it attempts to incentivize users to seek medical attention when necessary, and acts as a pre-screen similar to how at-home pregnancy tests act as a pre-screen for a doctor's check-up.

2.1.2 Optional Advanced Imaging Techniques

In addition to a visual examination, an increasing number of doctors are using a technique known as dermatoscopy to help identify spots on the skin more easily. The dermato-

*Advisor: Zachary Ives(zives@cis.upenn.edu).

scope consists of a special magnifying lens and a polarizing light source that allows for closer inspection.[6] Doctors often take digital photos of the spots as well. An experienced eye aided by a dermatoscope is proven to have increased accuracy over naked eye observation and enhanced differential diagnosis.[7]

Besides dermatoscopy, Spectrophotometric Intracutaneous Analysis System (SIAscopy) and MelaFind are devices that emit radiations ranging from 400 to 950 nm into the skin to obtain a variety of clinical and dermatoscope images.[8] Both techniques measure the amount of light reflected for each of the wavelengths, which is useful to observe because the optical properties of skin components tend to differ. They absorb or reflect light in varying quantities and may react preferentially to certain wavelengths. SIAscopes analyze information about the location, quantity and distribution of melanin, collagen and haemoglobin in the skin using pattern-recognition algorithms. This is then presented to the dermatologist on a screen in graphical form known as SIAGraphs. The SIAGraph for someone with melanoma has been established and is easily identifiable. SIAGraphs therefore allow doctors with limited experience in the field to identify melanoma in its early stages.

MelaFind not only creates graphs similar to SIAGraphs but also analyzes these graphs to provide an entirely automated diagnosis.[9] However, the diagnoses provided by using SIAscopes and MelaFind tend to result in a lot of false positives and require a biopsy to confirm the presence of melanoma.[10]

Our solution can be seen as more of a pre-screening process to SIAscopies and MelaFind. It is not able to provide graphs similar to SIAGraphs as mobile phones are not capable of emitting light of different wavelengths. However, it can reduce the number of unnecessary MelaFind tests and SIAscopies and incentivize those who do need them to seek medical attention sooner. Receiving a SIAscope examination costs the patient \$250.[11] MelaFind costs \$7,500 on average and each session costs \$150 to \$200, and requires the presence of a doctor for administration.[12]

2.2 Existing Mobile Applications

At present, there are several mobile phone applications in the skin cancer detection space. However, our application differs from and exceeds these in various ways. Below are two of these applications.

2.2.1 SkinVision

SkinVision is an application that allows users to upload images of their moles and helps them track their moles' evolution. It also helps users understand the symptoms and risk factors associated with skin cancer.[13] Our application is more diagnostic and goes beyond tracking to actually inform users of their probability of having skin cancer.

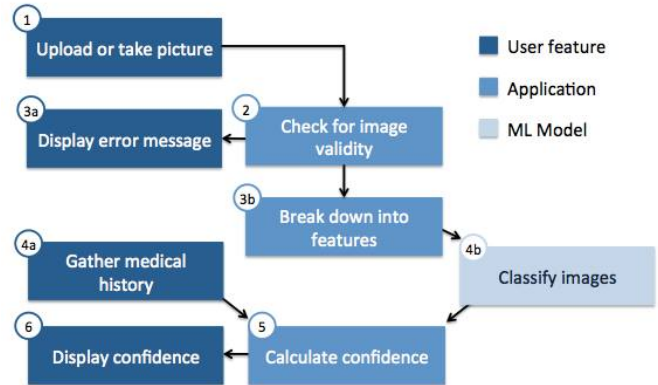
2.2.2 Doctor Mole

Doctor Mole provides analysis of an image uploaded by the user instantaneously. It analyses the image based on the asymmetry, border, color, diameter, and evolution criteria of evaluating a moles and reports back numbers for each criteria independently. The application makes use of a "freemium" model and provides more detailed analysis for users who pay for it.[7] Our application is an improvement over Doctor Mole by providing a consolidated probability

that takes integrates many features that dermatologists consider.

3. SYSTEM MODEL

Figure 1: System block diagram



Mole Investigator is our iOS application that allows users to take pictures of their moles to determine if they are at risk for melanoma. The application combines an intuitive and user-friendly layout with a powerful machine-learning based backend to help users monitor and analyze their skin moles that may be at risk for melanoma. As shown in Figure 1, it consists of five major components including the user interface, image validity checker, the machine learning model, and the user questionnaire.

1) Upload or take a picture: Mole Investigator, like many mobile applications, allows users to both take a new picture of a mole within our application or upload a picture to the application that they have captured in the past and saved to their device's photo log. Because iOS saves all images as JPEG files in the Photo Stream [14], it allows Mole Investigator to standardize image processing and loading to our machine learning model for only JPEG files.

2) Check for image validity: This is an important step in our application for usability purposes. What if a user accidentally uploaded a picture of their cat to our application instead of a picture of their skin mole? The image validity should also be quick to return an error in case the user did make an error. A user should not have to wait for more than 3 seconds to see if their image passed our validity checker. As a failsafe in case our validity checker does not work correctly, the original image is displayed again in the results page of the application.

3a) Display error message: If the user does input an invalid page, the application redirects them back to the original landing page with an error message at the top of the window.

3b) Break down into features: Based on our conversations with Dr. Jeremy Etkorn and Dr. Emily Chou at the Perelman School of Medicine[4], we were able to determine the specific visual features that doctors assess during a pre-screen of a patient's skin mole. While there are many risk factors and visual indicators that doctors assess, the core qualities that doctors observe are the asymmetry of the mole, the border of the mole, the color of the mole relative to the patient's natural skin color and within the mole itself, whether the mole is enlarging over time, the amount of sun

exposure the patient has had, and if the mole is significantly different from other moles on the patient’s body. The latter three features are difficult to determine from a still image because of the absence of change over time in a static image, the fact that amount of sun exposure is not visible in a mole, and the lack of other moles in the picture, respectively. After the image is sent to our server from our application, it is loaded into our backend code using OpenCV [15], a native Python library for analyzing and manipulating images using standard computer vision algorithms. OpenCV loads in images as two-dimensional numeric arrays, which is optimal for quick computations on the machine learning model that we have trained.

4a) Gather medical history: The need for a medical history section of our application become readily apparent from our conversations with doctors. While the physical qualities of the mole are undoubtedly important, the medical history helps establish the context in which doctors base their initial assumptions and suspicions. Currently, our medical questionnaire evaluates four factors: whether the mole has been enlarging over time, whether the mole looks significantly different than other moles on the user’s body, the user’s age, and the user’s history of sun exposure through different periods in their life. The full user questionnaire is given in Figure 3.

4b) Classify images: The machine learning model is saved as a pickled file on the server’s filesystem. The server reads in the image features and then passes them to a the `predict()` method on our regressor, which calculates a confidence measure between 0 of 1 (where 1 represents a mole that is definitely benign).

5) Calculate and Display Confidence: The regressor score of the machine learning model is weighted as 80% of the final confidence measure, and the results of the questionnaire is weighted as the remaining 20%. This distribution was determined by the rough weightage that doctors told us they use when determining the status of a mole during a pre-screen. The final confidence interval is then displayed to the user, along with a recommendation to visit a doctor if their score is very low and a medical disclaimer that ensures that our application is not interpreted as a substitute for a medical professional’s advice.

4. SYSTEM IMPLEMENTATION

4.1 User Interface

Mole Investigator’s user interface is designed to be intuitive for a first-time user of the application. A welcome screen simply asks the user to press a “Start” button that sends the user directly to the first question in the medical history questionnaire. In contrast, most mobile applications involve a nontrivial onboarding process for a new user that generally involves creating an account or logging in to an existing account. Due to privacy concerns, Mole Investigator does not involve the creation of a user account or any user identification. It is designed as a single-use application that analyzes an inputted image but does not store the image permanently on any server or remote host after the application finishes outputting a classification score.

The user first passes through four different interfaces during the medical questionnaire. Each interface contains exactly one question from the questionnaire, each with accompanying image metadata that allows the user to understand

exactly what is being asked. For example, the first question asks the user if their mole has been enlarging over time. Above the question, a diagram that portrays a progression of a enlarging mole allows the user to compare their mole’s growth and current status to the example mole in the image. These images also give the interface a sense of legitimacy and polish that is lacking in the comparable applications that were on the market in the past.

In the last step, the user interface allows the user to upload an image or take a new picture of a mole on their body. The image is then submitted and a confidence interval is displayed based on the output of the machine learning regressor and the answers to the questionnaire. If a user’s score falls below 60%, a message is displayed that reads “Your mole image exhibits risk factors commonly associated with melanoma. We recommend visiting a doctor for an examination.” This message does not directly tell the user that they may have cancer, but it also forcefully directs them to see a physician due to a potential urgent medical condition. A disclaimer is also displayed in the end screen, regardless of classification, that informs the user that the application does not claim to be substitutes for real medical professionals and the output of the models is subject to error. This message is placed for legal reasons and would exist in any official release of the application.

4.2 Application

Mole Investigator uses the hosting service Heroku [16] to serve our API endpoint that the application calls when sending an image server. The endpoint is written using the popular Python microframework Flask. The use of Flask [17] allows the use of the Python programming language to write the entire backend of the application, which includes the image preprocessing and machine learning model. Python is an ideal backend to use for this project because of its fast numeric processing libraries and native packages for the machine learning library Scikit-Learn [18] and OpenCV, the two main libraries used for feature extraction. The Flask application contains two paths: `/valid` and `/data`. Both routes take one query parameter: `img`, which is passed to the route as a serialized JSON [19] object. As the name indicates, the `/valid` endpoint is used to assess if the object passed is a valid mole image. The `/data` endpoint is the endpoint that actually performs machine learning on the passed objects and outputs a confidence interval based on the features it extracts from the image. On the client side, the application simply loads in the image to the native iOS format, wraps it as a serialized JSON object using a standard library available to all Apple developers, and calls the Heroku endpoint with the JSON passed in as a query parameter. The use of two routes allows the application to avoid memory leaks and also separates the two main functions that the backend performs for the frontend, allowing for more effective debugging and work distribution.

4.2.1 Separating Background Skin from Mole Region

The first function that the application performs is checking whether the image is a valid image of a mole. This validity check must be accurate, but also fairly quick in order to enable a quick turnaround by the user if they do input a poor image. The use of the `/valid` route on the Heroku endpoint allows us to efficiently pass the image to a dedicated worker on the Heroku instance that runs the backend code

on the image that is passed in as a query parameter. On the backend, an analysis is performed to determine if a mole outline exists in the image passed into the endpoint. First, a k-means clustering on the image (with $k = 2$) captures the major components of the image and segments the image into the background skin color and the actual mole area. A customized method is also overloaded onto the existing k-means segmentation library that outputs all the pixels in the image that belong to one of the specific segments. An example of this segmentation can be found in Figure 2. The validity checker then checks to see if the mole region (which is generally on the interior of the image) is contiguous. This is achieved by running a breadth-first search starting from one of pixels in the mole region and assessing if 80% of the pixels that are marked as mole pixels can be reached by the search. One shortcoming of this approach is that a random pixel may be chosen that is not attached to the main body of the mole. To mitigate this issue, the search is run three times and an especially low search result is thrown out if the other two searches have values that exceed 80%. This validity checker could be fooled by images of different types with similar attributes, such as a plate on a dinner table. It is the area of the application that could use the greatest improvement in future iterations. To act as a failsafe, a picture of the original image taken is displayed in the end screen to show the user what they originally inputted. If they inputted the wrong image, but the image still passed the validity checker, they will see it in the end screen and redo the workflow once they realize their error.

4.3 Machine Learning Model

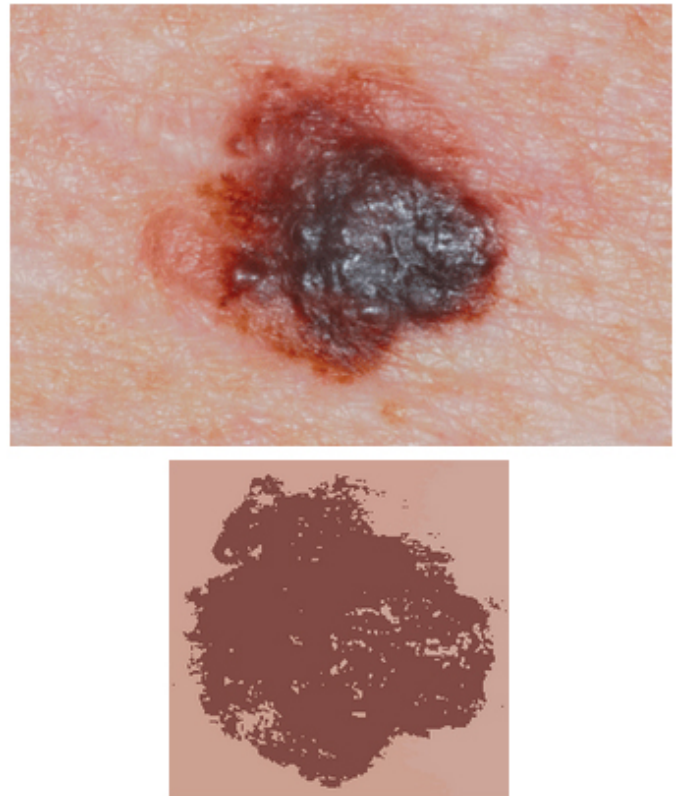
From our discussion with doctors, we verified the six key features that doctors analyze during a visual melanoma examination. Of these attributes, asymmetry, border and color can be extracted from the image. Evolution is difficult to extract from the image due to the limitations of one picture to demonstrate evolution. Ugly Duckling Sign and sun exposure are non-image related features that will be inputted at the user questionnaire level in the product workflow. These attributes cannot be used to build our machine learning model because our training images do not come with the necessary accompanying information. Recording these features from the start may allow us to incorporate these features into the model in the future given labeled data. The main technical challenge of this project, however, is effectively and consistently extracting features from the image provided.

Each of the three features that can be judged from a given image (asymmetry, border, and color) present a unique challenge from a technical perspective. The implementation and rationale behind the feature extraction processes will now be explained.

4.3.1 Preprocessing

As in many machine learning applications, data that are inputted into a model go through some sort of standardization that reduces error due to white noise in the model. While training the machine learning model, we threw out images that exceeded a 800×600 size because images larger than this size would have a resolution that could not be matched by an iPhone camera. The key standardization performed was using the same k-means segmentation described in the 'Application' section to create a bounding

Figure 2: Result of K-means Clustering on Malignant Mole ($K = 2$)



box around the mole and cropping the image to fit tightly around the mole region. This eliminated a large amount of white noise by removing large portions of the image that contained only skin pixels. The larger the portion of mole pixels in the analyzed image, the more effective our feature extraction processes were at inferring the different features within the image. The bounding box approach also helped center the mole within the cropped image, a key operation that helped with extracting the symmetry of the mole.

4.3.2 Extracting the Asymmetry Feature

Finding the measure of symmetry in an image is a computationally intensive problem that involves foreground/background analysis and inferring different shapes and patterns within an image. Fully implementing a symmetry algorithm would have made the application unusable because the inefficient runtime of any algorithm that could have been developed. Yet the format of the standardized mole image (centered image that contains two dominant segments) allows for greater assumptions to be made about the image.

First, the image is split into two halves: left half and right half. The right half is then reflected using standard OpenCV image manipulation libraries, which makes the two halves now directly comparable to each other by orientation. This does not invalidate any of the images because the standardization step centered the moles in the cropped image. The halves are then converted to black and white images to remove any discrepancies in color from the analysis of the symmetry feature. After these operations, the problem has been reduced from determining image symmetry (a hard

problem) to determining image similarity (a comparatively easier problem).

From here, two metrics are computed to represent symmetry in the feature matrix for each image. First, a pixel-by-pixel comparison of the two halves yields the mean squared error of the pixel values. This is a good heuristic for the image symmetry because it directly compares pixels across the reflected half and the original half; the pixels are also grayscaled, so the error is being computed from the pixel intensity value on grayscale. A value of 0 for the mean-squared error implies perfect symmetry, and larger values imply less symmetric images. However, the mean squared error is not a very sophisticated measure and is suspect to large fluctuations in one-off pixels, especially if the images are small. The SSIM image similarity metric, described by Zhang et. al [20], is a more robust metric because it can sense image structure and shapes to compare image similarity instead of simply comparing pixel values. The SSIM metric also accounts for the variance between pixel intensities, which mitigates the effect of any pixels that are drastically different from each other. This metric is assessed from a -1 to 1 scale, where -1 represents perfect dissimilarity and 1 represents perfect similarity.

4.3.3 *Extracting the Border Feature*

The border feature was the most academically interesting feature to extract. When doctors assess the border of a mole, they are looking to see if the border is jagged and irregular when compared to the smooth curvature of a benign mole. Therefore, we harnessed the power of the OpenCV Python library to fit curves to the mole and assess the error between the curve of best fit and the actual border that could be extracted from the mole.

Because moles can come in many different shapes, ellipses were the best way to fit on both irregularly shaped and regularly shaped moles. OpenCV had a contour plot function that fit around the border of the mole region of the image, and OpenCV was also used to draw the exact outline of the border around the mole. The residuals of the mole outline and the "ellipse-of-best-fit" were calculated as fractional scales of the entire image. For example, if a 200 x 100 image had a residual of 40 vertical pixels, then the error recorded in the border metric would be $40/200 = 0.2$. These residuals were added and averaged across the size of the border to get an average border residual measure, which is a heuristic for how jagged and irregular the border was compared to a perfectly smooth elliptical curve.

The border feature extraction worked well across across images of all types and shapes because it relied on tried and tested OpenCV fitting libraries. 98% of images in the training set were able to output a border and a perfect ellipse when analyzed using the border tool. One issue noted is the OpenCV contour fitting method had inconsistencies when fitting curves on images that had a darker skin color or a dark skin color that closely matched the color of a dark mole. In general, a distribution of skin colors that was representative of the population was lacking from our training set because of the limitations of finding raw images of skin moles on the Internet that are of the correct size and resolution.

4.3.4 *Extracting the Color Feature*

A high contrast between the color of the mole and the

skin color of a user is a risk factor commonly associated with melanoma. Yet doctors also evaluate whether the mole contains multiple primary color segments, which indicates that the mole is undergoing rapid chemical and structural changes that could lead to melanoma. The color feature developed attempted to balance these two qualitative observations into one unbiased metric.

First, the standardized image was clustered into three primary color segments using another k-means segmentation with $k = 3$. Yet we did not use a standard k-means segmentation Sci-kit learn library because of its relative inefficiency. Since we were more interested in the actual colors and not the segments, a custom program on the backend was written that simply tracked the pixel values of the centroids after the segmentation was run instead of actually attempting to segment every pixel into a specific cluster. The three primary colors was significant because it would include the primary skin color, the primary mole color, and a secondary mole color. If a mole was largely only one color, then the RGB values of the primary and secondary mole color would be very similar, indicating a benign mole. Conversely, if the RGB values of the two mole colors differed greatly from the RGB value of the skin color, this would indicate a high contrast between the mole's color(s) and the skin color. If all three color values were different, then this would be a glaring indicator that the mole was at the very least abnormal and should be examined by a doctor.

The evaluation metric used for color was a function of the differences between the RGB values of the three primary colors in the image. The RGB color standard is naturally mapped out across three dimensions (red, green, and blue), and each value is encoded in 8 bits (0 - 255 in base 10). It was then simply a matter of calculating the "distance" between each color using standard Euclidean distance and then averaging these distances together to create the total average color distance for the three primary color. As with the border feature, this metric performed inconsistently on our limited samples that contained subjects with dark skin color, as their moles' colors tended to be similar to their skin color simply because their skin color was dark. One way this issue was mitigated was calculating the total average skin color in the image and weighting the average color distance more heavily towards color discrepancies within the mole than to differences between the mole and the skin. If this was not performed, then users with darker skin color would be given a false sense of security by a low color contrast score.

4.3.5 *Building the Machine Learning Model*

Our initial training set of 200 images (100 benign, 100 melanoma) were gathered from the Internet by simply using Google search to find different types of moles, assessing if the images were the correct size and resolution as images that are taken on the iPhone. Because this was very time-intensive process, our training set contains images of good quality that are not representative of the general US population simply because it was difficult to find images that contained a specific skin color. Most of the training images were taken of people of Caucasian descent, but there were 10-15 images of moles with darker skin color.

After extracting the features from the images, a specific type of machine learning regressor had to be chosen to segment the images. Many different types of models were tested,

such as Gaussian regression, Bayesian modeling, linear and polynomial SVMs, and logistic regression. All of these models are built into the default Scikit-learn Python library. Ultimately, the random forest regressor had the highest F-score of any of the models and also the highest precision, which was important because we wanted to bias towards a high precision. The random forest regressor is an example of an ensemble method that combines the predictions of several base estimators with a given learning algorithm to improve the generalization of results. The random forest regressor uses several decision trees and averages over all of their outputs to reduce overfitting and improve the accuracy of the overall model [21]. It is a good choice for our dataset because it can ignore white noise that comes from an unstandardized dataset.

Because image data is unstandardized by nature, and moles can come in all different shapes and colors, the random forest regressor helped filter through many of the issues commonly associated with handling image features. The two parameters that needed tuning within the random forest regressor was the maximum depth of the decision trees and the number of estimators in the forest. GridCV is a tool built into Scikit-learn that allows a developer to find the optimal parameters for a specific metric. The optimal parameters for the regressor was 30 estimators and no pruning, so max depth was set as "None". There are many other parameters that could have been tuned, but they did not change significantly in the grid search.

After the regressor was trained on the 200 training images, it was dumped and saves as a flat file on the Heroku endpoint. When images are passed to the endpoint, the file is opened and read in as a machine learning model using the Python module Pickle [22]. Another advantage to tuning only two parameters is the small file size of the dumped model, which makes the image processing much faster. The regressor then fits to the passed image and outputs a score between 0 and 1, where 1 represents a benign mole. If the score falls below .6, then the application considers the image as melanoma. Otherwise, the mole is classified as benign. This binary classification is the basis of the calculation of our summary statistics, such as precision, recall, accuracy, and F-score.

4.3.6 Biasing towards Precision

Like many machine learning models, there was a natural tradeoff between precision and recall. Mole Investigator is biased towards having a higher precision because the application should not cause a user to panic and think that they have melanoma unless many of the features exhibited by the mole are in line with common risk factors associated with melanoma. Having a lower recall score is an acceptable tradeoff, as users would probably self-check themselves regularly, increasing the odds that a false negative in one self-check would then be confirmed as a positive image in a future self-check. The random forest regressor also inherently reduces the variance in the model at the cost of having a higher bias, which is actually also an advantage for our application.

4.4 Questionnaire

To address the three factors mentioned in the previous section that cannot be gleaned from the image analysis, we included the following questionnaire in our application:

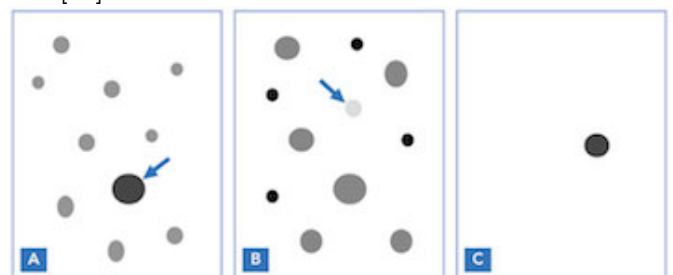
1. Has the mole changed in appearance?
Options are a) No b) Somewhat c) A lot

Figure 3: Users are provided with an example image that shows them what an evolving mole looks like. [23]



2. Is this mole noticeably different in appearance from other moles on your body?
3. How old are you?
4. During each of these periods of your life, how much time did you spend in the sun on average per day?
 - (a) 0-20 years old
 - (b) 21-40 years old
 - (c) *So on...The last of the ranges provided is based on what the user inputs as their current age.*

Figure 4: Users are provided with example images showing them what an "ugly duckling" mole looks like. [24]



Based on our research with doctors, the "Ugly Duckling Sign" is the most important of these factors to consider, followed by evolution, and sun exposure respectively.[4] This information was used to determine how heavily each of these factors should be weighted in the determination of the likelihood of melanoma, eventually settling on the 80/20 split between the results of the machine learning regressor and the questionnaire, respectively.

5. RESULTS

5.1 Machine Learning Model Standards

5.1.1 Summary Statistics

The main statistics used to judge the efficacy of our model are precision, recall, and accuracy. The model was found to have a precision of 89.1%, a recall of 68.3% and an accuracy of 79.8%. This accuracy number is based on a 80-20% split of our dataset as training and test images. As mentioned earlier, our model was biased away from giving false positives in order to avoid panicking users unnecessarily. Therefore, high precision

$$\frac{\text{truepositives}}{\text{truepositives} + \text{falsepositives}},$$

was valued over high recall

$$\frac{\text{truepositives}}{\text{truepositives} + \text{falsenegatives}}$$

and our model aligns with this. Nonetheless, we did not want to overcompensate for high precision and end up with a low recall model. Consequently, the F-score, which is a metric commonly used in machine learning that represents an aggregation of both precision and recall, was also analyzed.[25] Our model's F-score was found to be 77.3%, which falls within an acceptable range. Based on the design of our model, these metrics will only improve as the dataset gets larger and more robust.

5.1.2 K-fold Cross Validation

The next evaluation step involved performing a k-fold cross validation, with ten folds. The entire data set was divided into 10 parts. Our classifier was then trained on nine of these parts and tested on the part that was left out. In this fashion, 9 more supervised learning models were performed in which a new "fold" of the data set served as the test data for each iteration. Summary statistics for each fold were then calculated along with the mean across folds. The mean accuracy was found to be 75.1%. The k-fold cross validation is a rigorous test that helps prove that our model continues to perform well given different images in the dataset.

5.1.3 Efficiency

The time our model takes to train and test was also measured to ensure that our model is efficient. While this performance standard is secondary to the performance of the precision and recall of the machine learning model, it is an important consideration to make our model extensible and usable beyond our localized backend. While testing our model locally, we observed a processing time of 2.79 seconds per image. Currently, our basic k-means clustering algorithm runs in 1.47 seconds, but using OpenCV to infer shapes in the mole adds an additional 1.32 seconds to our run time.

5.2 Mobile Application Standards

5.2.1 Classification to Probability

Because Mole Investigator does not output a blatant "Yes/No" to users, our model's findings and the questionnaire results had to be translated into a probability to be returned to the user via our mobile application. Judging the performance of this probability was difficult because of a lack of labeled

data from the user. To understand how representative the probability is, our application was manually tested on several of our family members and friends to ensure that the majority of the results lean towards a high probability of being benign. In all the tests, the probabilities returned indicated that the moles were 70-95% benign. However, we did not gain authorization to test our application on real cancer patients' images and answers.

5.2.2 Efficiency

After developing our mobile application and linking it to our backend model, there was some natural latency because of the series of GET and POST requests that our application issues and receives while interacting with the machine learning model. Once the user selects a photo to upload, it takes the application an average of 4.68 seconds to output results to the user for a large image and 3.94 seconds for a small image. During this time, the model is breaking down the image into features and classifying it. With a loading screen presented to the user in the meantime, this latency seems reasonable.

5.2.3 Ease of Use

Ease of use of our application was a very important goal for this project as we want to encourage more people to screen their moles. Ease of use was tested by asking our friends and families to try our application and verifying that they were able to use it smoothly without any help. All 32 people queried found the application very straightforward to navigate through and use.

6. FUTURE WORK

One the biggest impediments to improving the accuracy of the machine learning model beyond 79.8% was the lack of a robust dataset of images. The images used for the model were downloaded from the internet and may not have been an accurate representation of the population. Ideally, images would have been obtained from dermatologists' existing datasets. However, the legal process for doing so would have taken over 12 months. If this project were to be continued, we would go through the process to obtain this robust dataset to improve the accuracy of the model.

A second major improvement would be the further standardization of the user-taken images. The current application adjusts for the zoom-level but does not sufficiently account for the angle and lighting of the image. This would have required more advance computer vision techniques such as edge detection using Hough transform and perspective correction. Instead, users are supplied with a quick guide on how to take the mole image correctly. However, having a standardized image could further enhance the model's performance.

A third improvement would be to investigate the current medical privacy laws further. Currently, as discussed in the following Ethics and Legality section, Mole Investigator erred on the side of caution when it came to medical privacy laws. If we were to take this project further, we could hire legal counsel to help us determine how we can maximize our impact without violating the laws. This could allow us to store images of the user's mole and track the evolution of the mole. The mole's evolution was cited by dermatologists as a major factor in determining the likelihood of melanoma and is currently being included in the application's question-

naire. However, by allowing the user to take pictures over time the application will not only be more user-friendly but also contribute to the accuracy of the output as users may not have paid attention to changes in appearance.

Lastly, if the application were to go through the FDA approval process and be commercialized, it would require more thorough testing on users with benign and melanoma moles. Since none of our family or friends have melanoma, we would have to gain access to real patients which would require clearances that would have taken too long to obtain.

7. ETHICS AND LEGALITY

There are a few key ethical or legal factors that needed to be considered when building the application. Health related products are taken very seriously by consumers and organizations. The first ethical consideration was to not create overly panicked users. If users were to receive feedback that their mole is extremely likely to be melanoma, they could take this as a death sentence or become panicked and irrational. To avoid this, our model was biased away from giving false positives, as explained above. Conversely, we do not want to create complacent users. In order to create empowered, action-seeking users, the application urges the user to seek medical attention for any score below 60% benign, and mentions that users should consult doctors if they have any questions or concerns for any score.

A second concern, that is both ethical and legal, was to not mislead users into thinking that the application provides a firm diagnosis. In terms of ethics, it would not be ethical to give users the impression that the application can stand in for a doctor because that would hurt, rather than help users' chance of survival. In terms of legality, a skin cancer detection application, Mole Detective, was sued by the Federal Trade Commission for misleading users into thinking it was firmly diagnosing skin cancer.[26] In order to avoid this ethical and legal problems, careful disclaimers were put into the application to inform users that it is not a substitute for doctors.

A third concern was complying with the Health Insurance Portability and Accountability Act (HIPAA). HIPAA, among other things, "Protect[s] the confidentiality and security of healthcare information." [27] In order to avoid any possible regulatory infringement, the application does not store any of the images or data input by users. Many comparable medical applications for mobile devices involve the creation of user profiles, which the application also does not solicit.

8. CONCLUSIONS

We were able to accomplish our goal of building a mobile application to estimate whether users' moles are cancerous in order to incentivize at-risk users to seek medical attention. The application utilizes a random forest machine learning model that evaluates a user's mole image on the features of asymmetry, border irregularity, and color irregularity. Having been trained on a set of 100 melanoma and 100 benign mole images, the model is able to estimate whether the mole in question is melanoma, and performed with an accuracy of 79.8% on the test set. A questionnaire in the application gathers accompanying information about the user to refine the estimate. The estimated likelihood of melanoma is displayed to the user along with a message incentiviz-

ing seeking medical attention. The features analyzed in the model and questionnaire were developed through conversations with dermatologists and are meant to mimic a dermatologist's examination of a suspicious mole. By creating empowered and conscientious patients, this application is an exciting stride in the movement to give patients control of their health.

9. REFERENCES

- [1] W. H. Organization, "Skin cancers." <http://www.who.int/uv/faq/skincancer/en/index1.html>.
- [2] S. C. Foundation, "Skin cancer facts." <http://www.skincancer.org/skin-cancer-information/skin-cancer-facts>.
- [3] LiveScience, "Melanoma: Symptoms, treatment, and prevention." <http://www.livescience.com/34783-uv-rays-increase-melanoma-skin-cancer-risk.html>.
- [4] D. L. N. Dr. Emily Chu, Dr. Jeremy Etkorn, "Interviews."
- [5] S. M. Daniel, "Are there other methods of diagnosis, without biopsies?." <http://www.skincancer.org/skin-cancer-information/ask-the-experts/are-there-other-methods-of-diagnosis-without-biopsies>, 2013.
- [6] A. C. Society, "How is melanoma skin cancer diagnosed?." <http://www.cancer.org/cancer/skincancer-melanoma/detailedguide/melanoma-skin-cancer-diagnosed>, 2013.
- [7] M. Vestergaard, P. Macaskill, P. Holt, and S. Menzies, "Dermoscopy compared with naked eye examination for the diagnosis of primary melanoma: a meta-analysis of studies performed in a clinical setting." <http://onlinelibrary.wiley.com/doi/10.1111/j.1365-2133.2008.08713.x/abstract;jsessionid=194D7AA469DC.E59C5F1CA6E910D5ACB7.f01t01>, 2008.
- [8] M. Michalska, G. Chodorowsha, and D. Krasowska, "Siascopy - a new non-invasive technique of melanoma diagnosis." <http://www.ncbi.nlm.nih.gov/pubmed/16146122>, 2004.
- [9] G. M. Lipper, "Melafind for melanoma detection: Will it change practice?." <http://www.medscape.com/viewarticle/756436>.
- [10] C. Weaver, "Apps aim to detect skin cancer." <http://online.wsj.com/news/articles/SB100014241278//873>, 2013.
- [11] A. R. Cukras, "On the comparison of diagnosis and management of melanoma between dermatologists and melafind." <http://archderm.jamanetwork.com/article.aspx?articleid=1688057>, 2013.
- [12] M. McFadden, "Melafind: The cancer scanner." <http://www.wndu.com/mmm/headlines/Melafind-The-Cancer-Scanner-187932291.html>, 2013.
- [13] SkinVision, "About skinvision." <https://skinvision.com/about-skin-cancer-melanoma-mobile-app>.

- [14] J. Martellaro, "Why your iPhone uses PNG for screen shots and JPG for photos."
<http://www.macobserver.com/tmo/article/why-your-iphone-uses-png-for-screen-shots-and-jpg-for-photos>, 2014.
- [15] "The Opencv Reference Manual."
<http://docs.opencv.org/opencv2refman.pdf>, 2014.
- [16] "Heroku." www.heroku.com, 2015.
- [17] "Flask: Web Development, one drop at a time."
<http://flask.pocoo.org/>, 2015.
- [18] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python."
<http://www.jmlr.org/papers/volume12/pedregosa11a/pedregosa11a.pdf>, 2011.
- [19] "The JSON Data Interchange Format."
<http://www.ecma-international.org/publications/files/ECMA-ST/ECMA-404.pdf>, 2013.
- [20] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image Quality Assessment: From Error Visibility to Structural Similarity," 2014.
- [21] "Sklearn: Random Forest Regressors."
<http://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html>, 2015.
- [22] "pickle: Python Object Serializer," 2015.
- [23] N. C. Dermatology, "Types of mole."
<http://www.nyccosmeticdermatology.com/mole-removal.htm>.
- [24] N. C. Center, "Preventing skin cancer."
<http://www.nouveaucosmeticcenter.com/skin-screenings-skin-cancer-prevention-newark/>.
- [25] "Mean f score."
<https://www.kaggle.com/wiki/MeanFScore>.
- [26] B. Dolan, "Ftc fines two melanoma risk detection apps, melapp and mole detective."
<http://mobihealthnews.com/40770/ftc-fines-two-melanoma-risk-detection-apps-melapp-and-mole-detective/>, 2015.
- [27] "What is hipaa?."
<http://health.state.tn.us/hipaa/>.