STOR 565 Final Write-Up

Name: Max Van Fleet, William Petty, Vedika Birla, Claire Blackman, Ryan Helms.
Professor: Zhang
Date August 27, 2024

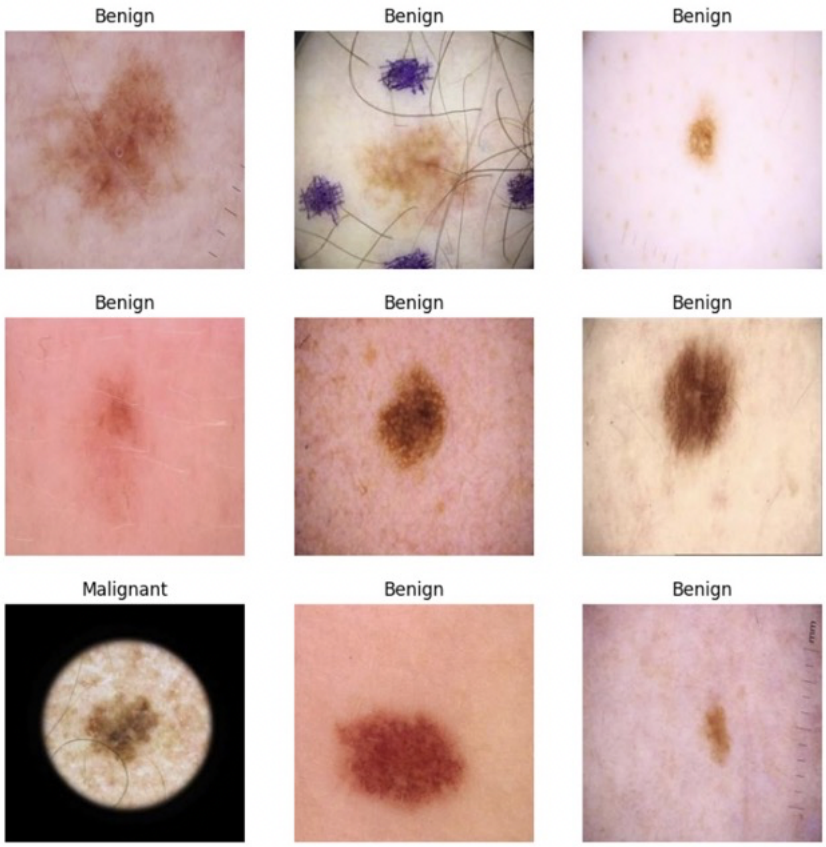
Introduction

Background

In this project, we are classifying melanoma cancer with transfer learning models. We were initially interested in working with healthcare data since there are many applications and significance to healthcare topics, especially diagnostics. This is an important topic, as over 100,000 melanoma lesions are expected to be diagnosed in 2024, and over 8000 people are expected to die from melanoma by the end of 2024 (American Cancer Society, 2024). By the age of 70, approximately 20% of Americans will be diagnosed with skin cancer (Skin Cancer Foundation, 2024). This project focuses on benign and malignant melanoma lesions where benign melanoma is not cancerous, while malignant melanoma is cancerous and can spread throughout the body (City of Hope, 2023). Early detection can lead to a survival rate of 99% (Skin Cancer Foundation, 2024). Our goal is to create a model where we can accurately classify melanoma images as benign or malignant. This would have significant implications for healthcare, specifically telehealth or at-home diagnoses. Benign tumors typically have smooth and defined borders (Cleveland Clinic, 2021). In contrast, malignant tumors do not, so we aim to create a model that can differentiate between these characteristics, ideally with some sort of feature extractor (Cleveland Clinic, 2021).

Dataset

Our *Melanoma Cancer Image Dataset* dataset was obtained on Kaggle from user Bhavesh Mittal. The data includes 13879 images of melanoma lesions, each 224 by 224 pixels. The data included a training and testing folder, each with subsets of malignant and benign melanoma images. Therefore, we can train our model on the training data and test our classifications on the test folder. There are 6289 benign and 5590 malignant images in the training folder and 1000 benign and 1000 malignant images in the test folder. There was no additional metadata other than the images themselves (Mittal, 2024).



Exploratory Data Analysis

In our exploratory data analysis, we attempted to use techniques we had been taught to better understand our data. We primarily employed Principal Component Analysis to perform dimension reduction and then analyzed the data in a dimension-reduced form. This approach was not ideal, as Principal Component Analysis is not well suited for nonlinear data such as image data.

PCA

First, we loaded the first 1000 benign and malignant images from our train set. We attempted to train some basic classification techniques on the PCA scores of these data, which were then tested on the first 50 malignant and the first 50 benign observations from the test set. We reorganized the $224 \times 224 \times 4 \times n$ image data into a two-dimensional, $n \times 150528$ matrix for principle component analysis. This was a naive approach, as our data failed to satisfy the assumptions of PCA, but at this point in the project, we had not learned more appropriate techniques. This simple approach was surprisingly effective, achieving a 65 percent test accuracy. However, closer observation revealed that this test accuracy was primarily driven by a 78 percent accuracy on the benign images, with only a 52 percent accuracy on the malignant. We experimented with fitting a support vector machine model to this data, weighting misclassifications of the malignant errors more heavily than benign errors. We experimented with several weightings, with 5:1 weighting generating the best results. This merely reduced our accuracy in classifying benign images but did not improve our ability to classify malignant cases. To conclude, our PCA

analysis showed promising results for separating the images based on malignant and benign melanoma lesions based on the plot, as seen by the fairly clear separation of the 2 classes. The first 3 PC values explained 65 percent of the variation, while the first 17 PC values explained 90 percent of the variation. However, PC is not the most appropriate method for this data, as the data is not linear. Many of these methods failed to classify malignant tumors appropriately, which is much more important than the accuracy rate for benign tumors.

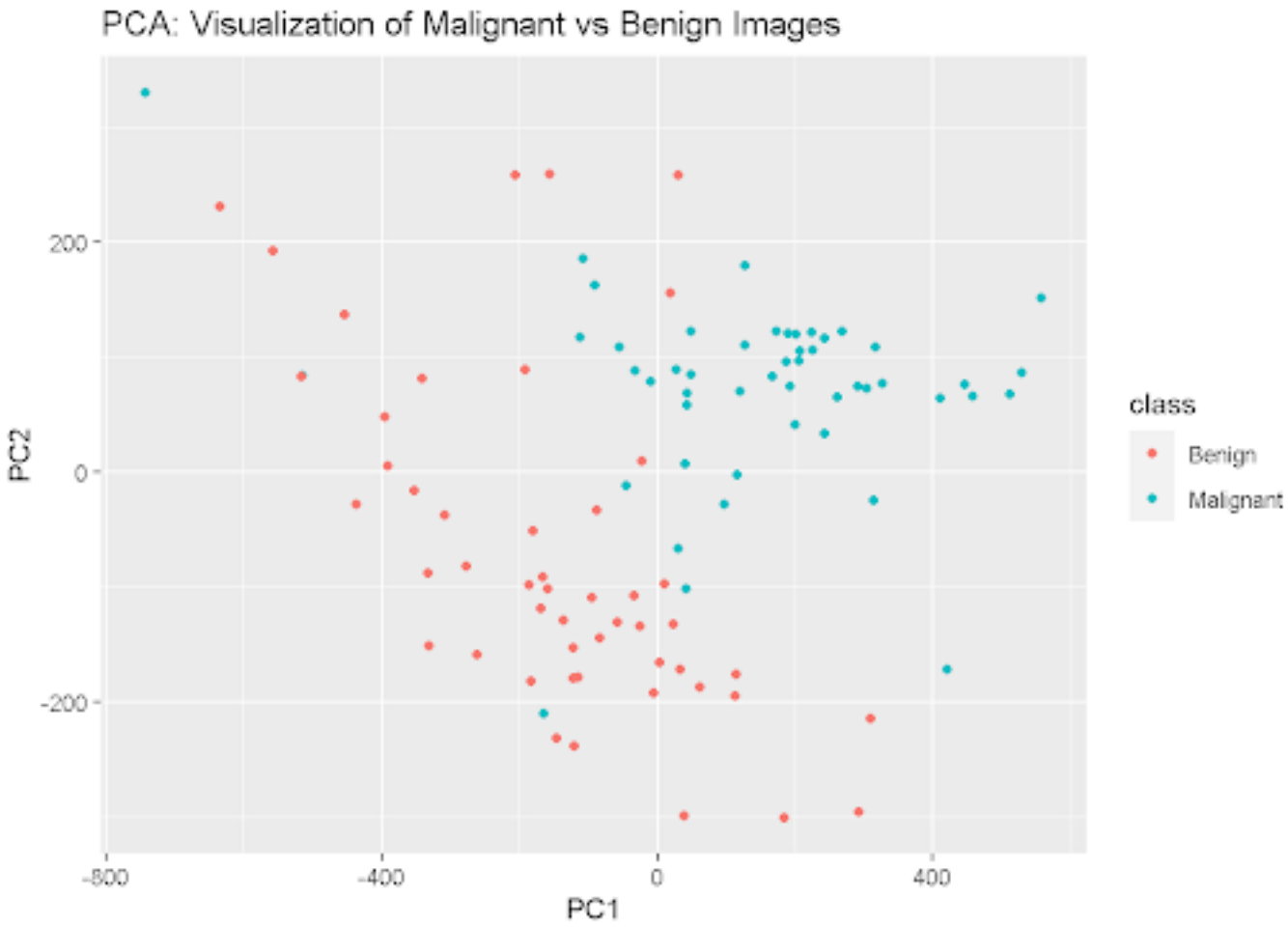


Figure 1: First Two Principle Components Graphed

Clustering

We employed a couple of clustering techniques to discover patterns and similarities in the data points and identify which machine learning methods would be most suitable for our dataset overall. First, we performed some basic k-means clustering analysis on various portions of the PCA scores, including the first 3, 17, 50, and 100 PCA scores. When choosing the K =2 and using the first 50 PCA scores, we achieved the best Adjusted Rand Index of 0.4298 ARI. Figure 2 displays the output of the K-means clustering algorithm run on the parameters above and graphed based on the first 2 principal components.

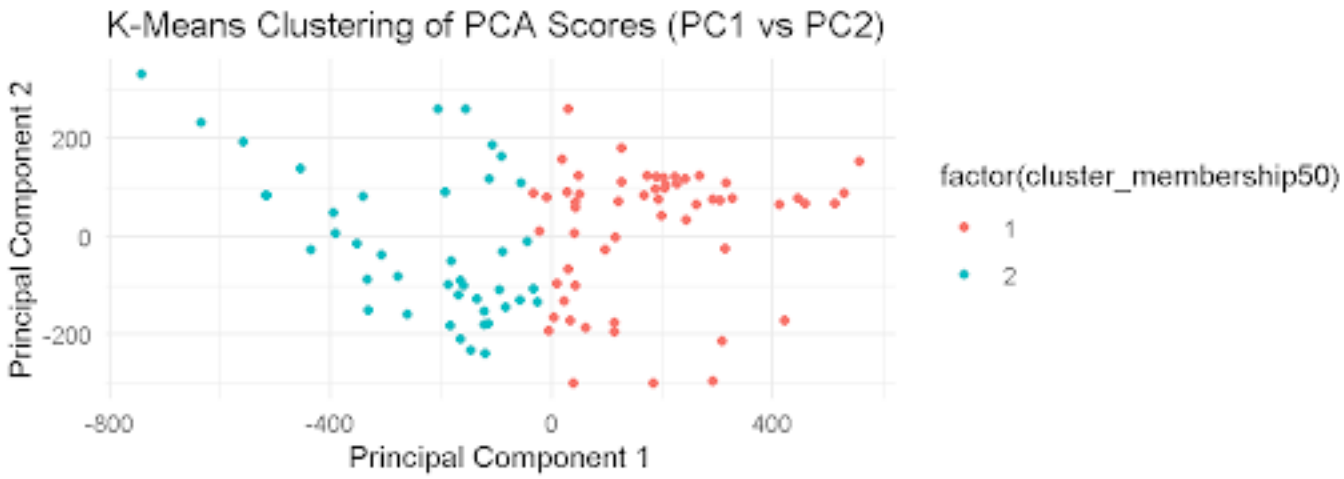


Figure 2: K-means Clustering using first 50 principle components and K = 2

We also experimented with some hierarchical clustering methods. Of these, Ward.D linkage performed the best by Adjusted Rand Index, achieving a 0.30796 in the best-observed repetition. The lesser performance of hierarchical clustering methods suggests that there is no hierarchical structure in our data, at least not one captured in the dimension reduced form. Below is the plot displaying the performance of Ward.D linkage hierarchical clustering.

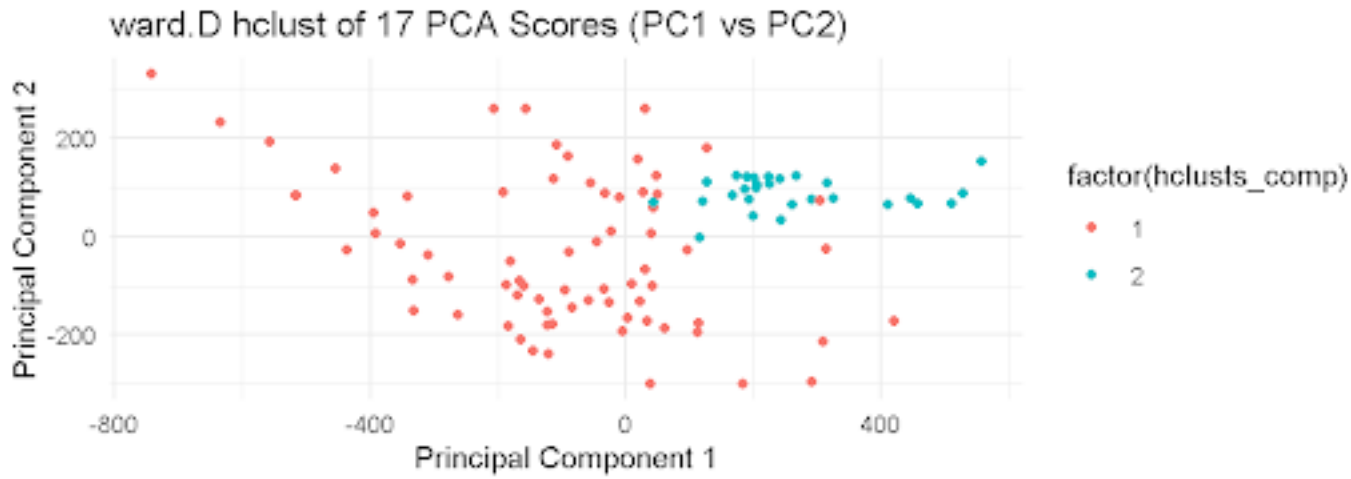


Figure 3: Best Iteration of hierarchical clustering using War.D linkage

Convolutional Neural Network

Convolutional Neural Networks (CNNs) are a deep learning algorithm mainly designed for image classification tasks. A convolutional neural network has four main elements: convolutional

layers, pooling layers, activation functions, and fully connected layers. Convolutional layers apply filters to input images to identify specific patterns like edges, textures, and shapes. Pooling layers pull out the most important features from the convoluted matrix by reducing the dimension of the feature map. They also help prevent overfitting. Activation functions are applied to introduce an element of nonlinearity into the model, allowing it to learn complex patterns in the data. They also help mitigate the vanishing gradient problems. Lastly, fully connected layers are the final layer of the CNN and are used to make final predictions. CNNs are trained using backpropagation, where the model fine-tunes its internal parameters based on the error rate to minimize the difference between its predictions and the ground truth labels in the training data (Keita, 2023). CNNs have been highly successful in various applications, including image classification, object detection, facial recognition, and medical image analysis, which is why CNNs are a suitable match for our project.

Methods

For our primary model we decided that a convolutional neural network would be the best option for a few reasons:

- Convolutional neural networks work well with higher dimensional data
- Convolutional neural networks can extract features that are in the data, which is helpful to deal with our highly non-linear data
- Transfer learning allows us to leverage the power of image classifiers trained on massive datasets, while using very little computational power of our own.

Initial Transfer Learning Step

Other methods, such as standard neural networks and SVM, will likely not perform as well as a convolutional neural would in this case due to the highly non-linear data and the high dimensionality of the data. For our base models, we employed Xception and EfficientNetV2B0 trained on Imagenet data (Fu, 2023; Chollet, 2023). These datasets do not have a close relationship to melanoma data, but we can still leverage the feature extraction capabilities of these models by implementing our own prediction layer. In both cases, our model structure included an input layer that passed output to the pre-trained base model, then added a global average pooling layer, dropout regularization, and a final dense prediction layer. We trained these layers in our first training pass while leaving the base model frozen as a feature extractor. We used Binary Cross-Entropy as our loss function and the Adam optimizer with a 1e-3 learning rate. Our Xception and EfficientNet models had a similar performance at this stage, with both achieving roughly 85% validation set accuracy. We did not evaluate these models on the test set at this stage.

Fine Tuning

For the fine-tuning step, we had two competing concerns to balance. First, our target task is very different from our base models' original task. This would lead us to train a greater portion of the base model in the fine-tuning step. However, our dataset is fairly small. This incentivizes us to train less of the base model in our fine-tuning step, as overfitting is a concern. In practice, we split the difference. For the Xception-based model, we unfroze the entire model before training at a lower learning rate of $1 * 10^{-5}$, while for the EfficientNet base, we unfroze the last third of the model and trained at a similarly low learning rate. The results from this

were surprising. While the Xception-based model improved to 88% accuracy on the validation set, the validation accuracy of the EfficientNet-based model declined marginally. Our test set results with the Xception-based model likely suffered from overfitting due to our best accuracy deviating from the expected .5, indicating fine-tuning the whole model on any neural net with our data set size would likely lead to overfitting.

Data Augmentation

This task is an ideal use case for data augmentation methods. These images have been captured at many different angles, leaving orientation irrelevant. We were free to apply various data augmentation steps without losing meaning in the resulting images, such as horizontal and vertical flips, zooms with padding, and rotations.

Results

The first model we performed transfer learning on was the Xception model. When testing our model with a threshold of 0.5, we correctly classified 937/1000 benign moles and correctly classified 830/1000 malignant melanomas. Additionally, when we tried adjusting the threshold to see how that would lower our chances of misclassifying a malignant as benign, we calculated the accuracy and the recall at each threshold, and the results can be found in Table 2. Lastly, we plotted the ROC curve with an AUC of .95, which is displayed in Figure 4.

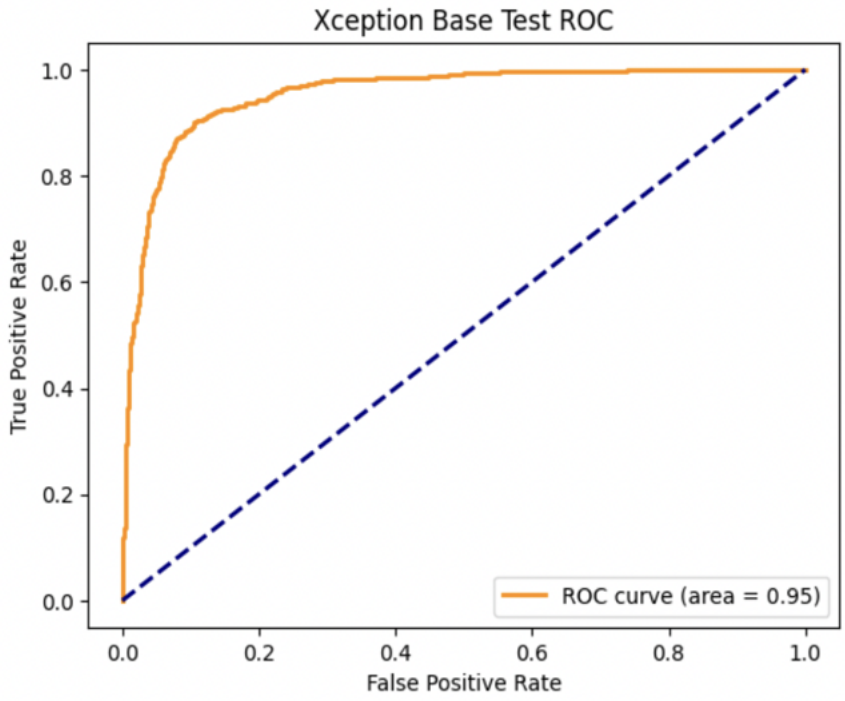


Figure 4: Xception ROC

	Actual Benign	Actual Malignant
Predicted Benign	937	170
Predicted Malignant	63	830

Table 1: Xception, Threshold .5

Thresholds	0.1	0.2	0.3	0.4	0.5
Recall	0.982	0.960	0.934	0.905	0.868
Specificity	0.552	0.723	0.804	0.858	0.894

Table 2: Recall-Accuracy Tradeoff of Xception Model

The Second model we performed transfer learning on was the EfficientNetV2B0 without any fine-tuning model. When testing our model with a threshold of 0.5, we correctly classified 949/1000 benign moles and 760/1000 malignant observations. Lastly, we plotted a ROC curve with an AUC of .96, which is displayed in Figure 5.

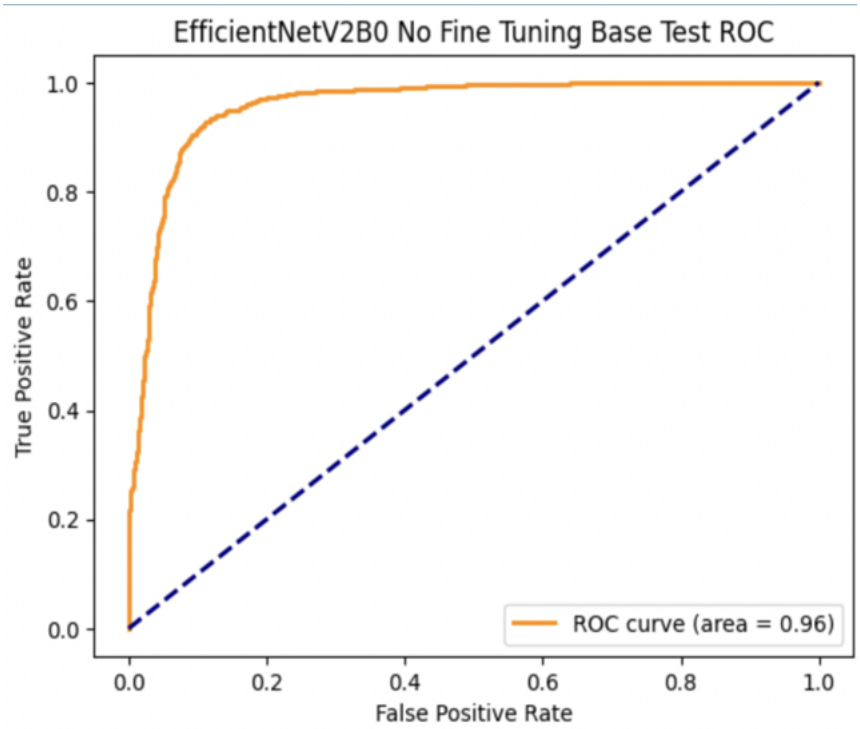


Figure 5: EfficientNet w.o. fine tuning

	Actual Benign	Actual Malignant
Predicted Benign	949	240
Predicted Malignant	51	760

Table 3: Confusion Matrix EfficientNetV2B0 With Out Fine Tuning, Threshold .5

The third model we performed transfer learning on was the EfficientNetV2B0 with a fine-tuning model. When testing our model with a threshold of 0.5, we correctly classified 946/1000 benign moles and 739/1000 malignant observations. Lastly, we plotted a ROC curve with an AUC of .94, which is displayed in Figure 6.

	Actual Benign	Actual Malignant
Predicted Benign	946	261
Predicted Malignant	54	739

Table 4: Confusion Matrix EfficientNetV2B0 with Fine Tuning, Threshold .5

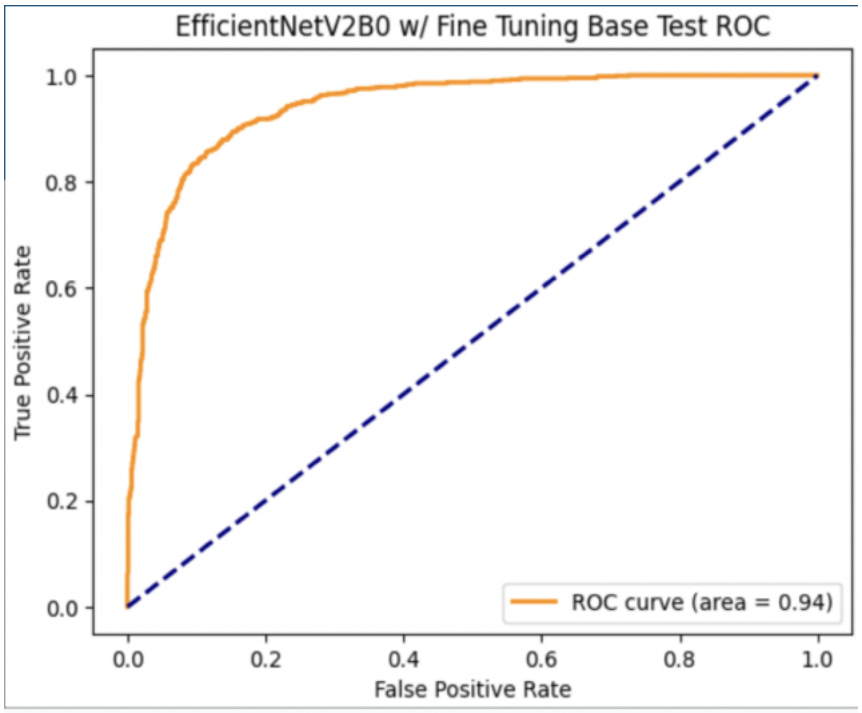


Figure 6: EfficientNet with fine tuning

Conclusion

Overall, our transfer learning methods worked reasonably well in recognizing these melanomas. This process highlighted the power and potential of transfer learning approaches, even for tasks

very different from the original tasks of the base model. This project was a good exercise for developing familiarity with transfer learning techniques, which were completely new to us.

Computational power was the main limiting factor in our analysis. Investing more computational power would open up many doors for further analysis of this data. A lack of computation rendered it infeasible to construct our own CNN from scratch, and a task-specific network could likely achieve similar accuracy with a much lower parameter count. More resources would also have allowed for a more robust comparison of different transfer learning techniques, such as experimenting with alternative base models and more experimentation with hyperparameters such as batch size and learning rate, in addition to the obvious benefits of allowing us to use a larger dataset, and more training time on the best performing models.

Works Cited

American Cancer Society. (2024, January 17). *Key Statistics for Melanoma Skin Cancer.* American Cancer Society. https://www.cancer.org/cancer/types/melanoma-skin-cancer/about/key-statistics

Skin Cancer Foundation. (2024, February). *Skin Cancer Facts & Statistics.* Skin Cancer Foundation. https://www.skincancer.org/skin-cancer-information/skin-cancer-facts/

City of Hope. (2023, January 12). *What's the difference? Benign vs. malignant tumors.* City of Hope. https://www.cancercenter.com/community/blog/2023/01/whats-the-difference-benign-vs-malignant-tumors

Cleveland Clinic. (2021, August 10). *Benign Tumor.* Cleveland Clinic. https://my.clevelandclinic.org/health/diseases/22121-benign-tumor

Mittal, B. (2024). *Melanoma Cancer Image Dataset.* Kaggle. https://www.kaggle.com/datasets/bhaveshmittal/melanoma-cancer-dataset

Keita, Z. (2023). *An introduction to Convolutional Neural Networks* datacamp. https://www.datacamp.com/tutorial/introduction-to-convolutional-neural-networks-cnns

Fu, Y. (2023, July 10). *Image classification via fine-tuning with EfficientNet.* Keras Examples. https://keras.io/examples/vision/image_classification_efficientnet_fine_tuning/

Chollet, F. (2023, June 25). *Transfer learning & fine-tuning.* Keras Guides. https://keras.io/guides/transfer_learning/