Final Report

Identifying Skin Cancer Using Skin Lesion Crops of 3D Whole Body Images That Resemble High-Quality Smartphone Images

By Lazaro Martull

Introduction

Skin cancer is a significant and growing healthcare concern, as highlighted by the Skin Cancer Foundation. Reports indicate that one in five Americans will be diagnosed with skin cancer during their lifetime, a statistic that is both staggering and alarming. Even more concerning is the mortality rate, with more than two individuals in the U.S. succumbing to skin cancer every hour. Melanoma, the deadliest form of skin cancer, accounts for the majority of fatalities. However, early detection can drastically improve outcomes; the five-year survival rate for melanoma is 94% when diagnosed before it spreads to the lymph nodes.

Despite these promising statistics for early detection, many individuals face barriers to accessing affordable diagnostic tools, often leading to late diagnoses at advanced stages of the disease.

The field of machine learning shows promising advancements in this area by offering an affordable option, for screening of skin cancer cases. In our study we explore the application of machine learning algorithms that have been taught using images of skin lesions taken from 3d full body scans with the intent to replicate the performance and convenience of top tier smartphone cameras. Our strategy is centered around closing the divide between expert evaluations and self-care practices empowering individuals to evaluate their health risks.

Through networks (CNNs) and transfer learning methods, our model's design process enhances the precision of categorizing and forecasting skin lesions under different imaging circumstances. Acknowledging the availability of smartphones well as their accessibility highlights their

potential as a valuable tool, for health-oriented purposes. A dependable AI powered resolution might motivate people to pursue expert assessment promptly which could potentially save lives by enabling intervention.

Our initial project aimed to address this issue by developing an AI model inspired by the ISIC 2024 Challenge to predict skin cancer from low-resolution images, such as those captured by smartphones. The ISIC 2024 dataset and the HAM10000 dataset were initially identified as key resources for training and validation.

We employed a transfer learning approach, leveraging pre-trained models such as ResNet50, InceptionV3, and VGG16 to optimize performance. After testing various architectures, we selected ResNet50 as the base model. The final architecture involved freezing the first three convolutional blocks of ResNet50, training only the fourth block, and appending custom VGG16-like convolutional layers. A dense layer with a “tanh” activation function preceded the final SoftMax activation layer for classification.

Moreover, this project tackles the hurdles of handling images from smartphones and the wider issues of bias in data processing and making sure the system is user-friendly and scalable. Differences in lighting conditions and skin tones in images can affect how well the model works so it is important to have methods and fairness checks in place. Moreover, our work highlights the significance of concerns by stressing that these tools should support professionals rather than act as a substitute for their ability.

In this report we discuss our efforts to create and deploy a machine learning system that can accurately detect the probability of skin cancer development. Our goal is to make early diagnostic tools more accessible, by concentrating on images taken with smartphones so that people can be more proactive about their health management. This initiative signifies a move towards using technology to tackle healthcare issues. Thus, potentially make a substantial impact on society.

Problem Statement

Challenge	Description
Image Quality Variability	Lighting conditions, resolution, and angle variations
Diverse Skin Tones and Lesion Types	Demographic differences and lesion variability
False Positives and Negatives	Overdiagnosis and underdiagnosis

Detection of skin cancer through smartphone-based tools offer potential for detection and improved accessibility in healthcare settings; however; these tools meet notable obstacles in terms of accuracy that hinder their widespread acceptance and dependability. Current models often face difficulties when dealing with real word scenarios where factors like image quality variations, lighting differences, and demographic diversity pose challenges not effectively addressed by the technology at hand. This lack of resilience may lead to diagnosis such as false positives or false negatives, thereby undermining confidence in these applications' effectiveness.

One major challenge lies in the consistency of images captured by smartphones as opposed to those taken in controlled settings where conditions are uniform and regulated for accuracy and quality assurance purposes; the varied lighting conditions and resolution levels, along with different viewing angles create disturbances and imperfections that could potentially confuse AI models and hinder their ability to accurately predict outcomes across diverse real-life situations. Moreover, the changing characteristics of skin abnormalities that can differ in dimensions, color, and form brings an added level of intricacy. Models need to differentiate between malignant anomalies while considering these disparities. Failure to do so could lead to negative results prompting unnecessary burdens on healthcare facilities, or false negatives, which could delay medical priorities.

Objectives:

This project sought to improve the precision and dependability of machine learning models for diagnosing skin cancer through smartphone images. To accomplish this, the following methods were implemented:

1. **Improving Smartphone-Based Skin Cancer Detection**: Improve the dependability and flexibility of machine learning models for identifying skin cancer using images taken with smartphone cameras.
2. **Image Quality Enhancement:** Utilize image processing methods to enhance the quality and uniformity of the input data.
3. **Fairness and Inclusivity**: Work on enhancing model fairness and inclusivity by implementing strategies to address biases and ensure performance for all groups of people.

4. **Model Efficiency**: Enhance model efficiency and cut down on training duration by leveraging existing trained models.
5. **Accurate Classification**: Develop a machine learning algorithm that can accurately categorize skin abnormalities across skin types and lesion features, within population groups.

Methodology

Datasets

After reviewing potential datasets, we determined that:

- ISIC 2024 Dataset: This dataset presented a significant class imbalance with over 400,000+ malignant images, making it challenging for model training.
- HAMM 10000 Dataset: While comprehensive, this dataset included only five types of malignant cancer and lacked benign images, rendering it unsuitable for binary classification.

Consequently, we pivoted to alternative datasets, including the more balanced ISIC 2018 dataset, which contained 519 malignant images and 2,075 benign images, for intermediate testing.

Image Preprocessing

To standardize the data and enhance image quality, we applied the following preprocessing steps:

- Sorting: Images were classified into benign and malignant directories based on the meta data files.
- Resizing: Images were resized to 224x224 to streamline the training process to ensure consistency and reduce computational overhead during training.
- Contrast Enhancement: Applied CLAHE (Contrast-Limited Adaptive Histogram Equalization) to improve local contrast and enhance image features, a technique commonly used in medical imaging.

Non-Clahe Image:

Re-sized CLAHE Image:

Model Configuration and Performance

Architecture:

1. Removing the 5th layer of ResNet50 which served as the base model.
2. The first three convolutional blocks were frozen, allowing only the fourth block to be trainable.
3. Custom VGG16-like convolutional layers and a dense layer (512 units) with tanh activation were appended.
4. A SoftMax activation layer was used for classification.

Training Configuration:

- Batch size: 8 for training, 16 for validation
- Early stopping with patience set to 20 epochs.
- Weighted the malignant class to 10x – to compensate for the data imbalance.
- Applied data augmentation techniques, including **RandomFlip, RandomRotation**, and **Rescaling**.

Testing Process:

- Initial testing was conducted using an Apples/Oranges dataset to ensure the foundational concepts of the model were sound. This dataset achieved a 98% recall score, confirming the model's ability to perform binary classification.
- The next phase of testing utilized the ISIC 2018 dataset, which provided a more balanced distribution of 519 malignant and 2,075 benign images. This intermediate step allowed further fine-tuning and validation of the model's performance.
- Final testing was performed on the ISIC 2024 dataset, which presented significant challenges due to its extreme class imbalance.

Unsuccessful Techniques

1) SMOTE (synthetic minority oversampling technique)
    a) Attempts to address the image class imbalance issue using SMOTE were ineffective as they generated synthetic images that failed to improve the model's performance. Furthermore, concerns were raised about the appropriateness of generating synthetic medical images.
2) Activation Functions:
    a) Testing various activation functions, such as ReLU, sigmoid, tanh, and SPLASH-based activation function caused validation losses to remain static across epochs, suggesting a possible implementation issue.

Pretrained Models

This section explores the concept of pretrained models and their specific purposes. A pretrained model is a machine learning model that has been trained on a large or sometimes massive dataset, allowing it to perform a specific task ("Pre Trained Model Definition | ENCORD," n.d.). Our goal was to successfully identify histologically confirmed skin cancer cases from photos based on the ISIC skin cancer detection competition on Kaggle (Kurtansky, Rotemberg, Gillis, Kose, Reade, Chow). The competition is over but will provide reasonable direction for our project and data from other competitors for comparison. An acceptable result would be >80% true positive rate.

How we used pretrained models: Our workflow chart

Below is a simple step-by-step table to list the steps in order of how we used a pretrained model:

1. **Choose a Model:** Select a pretrained model that aligns with your task and data.

ResNet50 was the pre-trained model that we chose – configured as described earlier.

2. **Load the Model:** Load the model's weights and architecture into your code.

   TensorFlow and Keras were the primary libraries that we used.

3. **Fine-tune (Optional):** If necessary, train the model further on your specific dataset to adapt it to your task. This process is called fine-tuning.

   The Adam optimizer was used with a training rate of .00001
   Batch sizes were set to 8 for training and 16 for validation
   Early stopping was enabled with a patience of 20 epochs

4. **Make Predictions:** Use the fine-tuned or directly loaded model to make predictions on new data.

   The predicted results are indicated in the pages below.

Workflow Chart: This model workflow was created using Microsoft Word with ideas from Kaggle ((Kurtansky, Rotemberg, Gillis, Kose, Reade, Chow).

Step	Description	Notes
1	Choose a Model	Consider model architecture, task suitability, and available resources.
2	Load the Model	Use appropriate deep learning framework libraries (TensorFlow, PyTorch, etc.).
3	Evaluate Initial Model Performance (Optional)	Test the model on a small sample of your data to assess its baseline performance.
4	Fine-tune (Optional)	If performance is unsatisfactory, train the model further on your dataset.
5	Make Predictions	Use the (fine-tuned or pre-trained) model to predict on new data.
6	Evaluate Model Performance	Assess the accuracy and other relevant metrics of your predictions.

The ResNet50 model (He et al., 2016) was used as a backbone for our image classification task below:

Code Used:

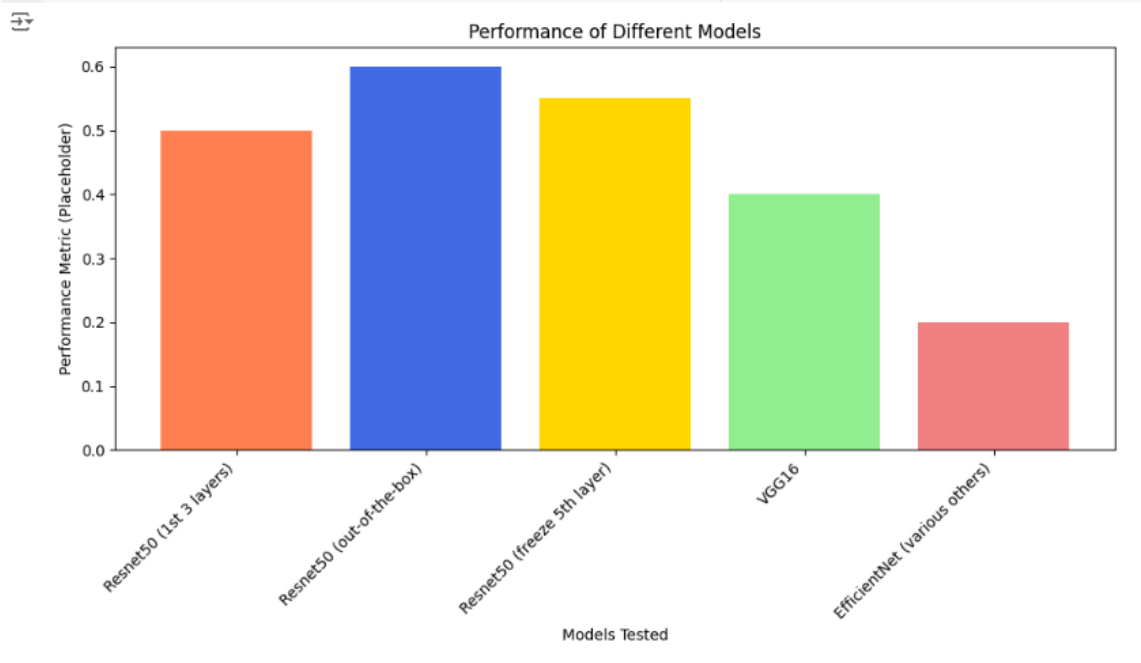
```
# Five Models tested
models = ["Resnet50 (1st 3 layers)", "Resnet50 (out-of-the-box)", "Resnet50 (freeze 5th layer)", "VGG16", "EfficientNet (various others)"]

# Performance (placeholder, replace with actual values if available)
performance = [0.5, 0.6, 0.55, 0.4, 0.2]  # Replace with your actual performance metrics

# Creating a bar chart with multiple colors
plt.figure(figsize=(10, 6))
plt.bar(models, performance, color=['coral', 'royalblue', 'gold', 'lightgreen', 'lightcoral', 'lightskyblue'])
plt.xlabel("Models Tested")
plt.ylabel("Performance Metric (Placeholder)")
plt.title("Performance of Different Models")
plt.xticks(rotation=45, ha="right")
plt.tight_layout()

# Display the chart
plt.show()
```

Output:



In summary, below is the clear picture of what the pretrained models mean using Google Sheets:

Model	Results
Resnet50 (first 3 layers)	Unsatisfactory
Resnet50 (out-of-the-box)	Over/underfitting
Resnet50 (frozen 5th layer)	Over/underfitting
Custom VGG16-like layers + Resnet50	Best results
VGG16	Limited testing
EfficientNet	Limited testing

The models' performance fell into these categories: unsatisfactory, overfitting, underfitting, best and those with limited testing. Unsatisfactory means not good enough, did not meet expectations. It also implied the results were below the required level of performance. Overfitting means the model memorized the training data including mistakes. It does great on the data it learned from, but poorly on new data. The full ResNet50, being a very complicated model, is more likely to do this, especially with limited data. Underfitting means the model was too simple to understand the data, so it performed poorly on both the data, and it learned from new data. Using only part of ResNet50 likely made it too simple. The best results were the ones that worked the best. Limited testing simply means not enough tests were done.

We tried using different parts of the pre-trained ResNet50 model, but it did not work well because it either learned the training data too well, also called overfitting, or not well enough, also called underfitting. Different configurations of the ResNet50 (full model, parts of it) all had this problem. They tried tweaking the model by only changing part of it, also called freezing the fifth layer, but this did not improve things and might have caused problems. In transfer learning, a pre-trained model like ResNet50 is used as a starting point. Instead of training the entire model from scratch, you often "freeze" some layers, meaning their weights remain fixed at their pre-trained values (Yuan et al.). Only the weights of the remaining layers are adjusted during training on your new dataset (Yuan et al.). Choosing which part to change is very important, and a bad choice can hurt the results.

Summary of Our Final Model

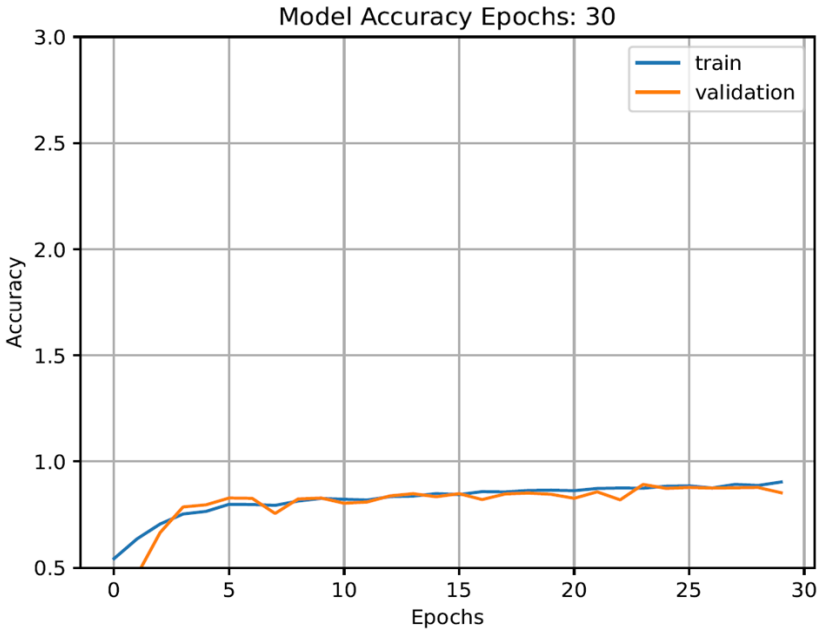
Model: Sequential

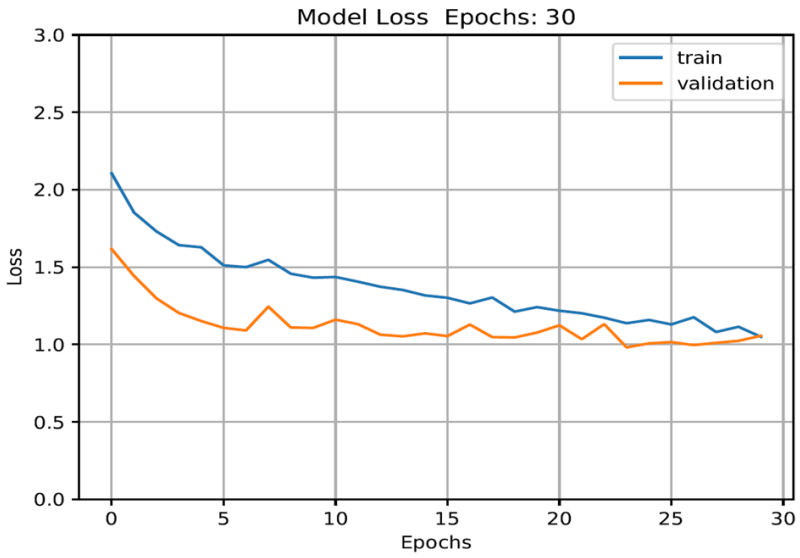
Layer (type)	Output Shape	Param #	
random_flip (RandomFlip)	(None, 224, 224, 3)	0	
random_rotation (RandomRotation)	(None, 224, 224, 3)	0	
functional (Functional)	(None, 14, 14, 1024)	8,589,184	- **This is ResNet50**
conv2d (Conv2D)	(None, 14, 14, 32)	294,944	
batch_normalization (BatchNormalization)	(None, 14, 14, 32)	128	
conv2d_1 (Conv2D)	(None, 14, 14, 64)	18,496	
batch_normalization_1 (BatchNormalization)	(None, 14, 14, 64)	256	
conv2d_2 (Conv2D)	(None, 14, 14, 64)	36,928	
batch_normalization_2 (BatchNormalization)	(None, 14, 14, 64)	256	
conv2d_3 (Conv2D)	(None, 14, 14, 128)	73,856	
batch_normalization_3 (BatchNormalization)	(None, 14, 14, 128)	512	
average_pooling2d (AveragePooling2D)	(None, 7, 7, 128)	0	
dropout (Dropout)	(None, 7, 7, 128)	0	
flatten (Flatten)	(None, 6272)	0	
dense (Dense)	(None, 512)	3,211,776	
dense_1 (Dense)	(None, 2)	1,026	

Total params: 33,719,784 (128.63 MB)
Trainable params: 10,746,210 (40.99 MB)
Non-trainable params: 1,481,152 (5.65 MB)
Optimizer params: 21,492,422 (81.99 MB)

Training Accuracy and Loss Results

The following graphs illustrate that model quickly converging with learning improvement and validation loss decreasing over 30 Epochs.





Test/Prediction Results

This is a non-visual or text-only results in confusion matrix and count:

- **Benign (benign) directory results:**
- **Benign (benign): 686 False Benign(benign): 91**
- **Malignant (malignant) directory results:**
- **Malignant (malignant): 53 False Malignant (benign): 21**

- **Benign accuracy: 0.8828828828828829 (Correct Benign Predictions)**
- **Malignant accuracy: 0.7162162162162162 (Correct Malignant Predictions)**

- **Accuracy: 0.8683901292596945 (Overall Correct Negatives (benign))**
- **Precision: 0.7162162162162162 (Overall Correct Positives (malignant)**
- **Sensitivity (Recall): 0.3680555555555556 (True Positive Measure (tp/(tp-fn)))**
- **Specificity: 0.7162162162162162 – (True Negative Measure (tn/tn+fp))**
- **F1 Score: 0.24311926605504586 – Overall Performance – Higher is better (2*(precision*recall)/(precision+recall))**

Prediction Results Visualization

	Predicted Benign	Predicted Malignant	Total
Actual Benign	686	91	777
Actual Malignant	21	53	74
Total	707	144	851

The display above, used with Microsoft Word to create the table, in the context of the spreadsheet design for organizing the classification results, is a table representing the confusion matrix and associated counts.

A confusion matrix summarizes the results of a classification prediction, showing the counts of correct and incorrect predictions for each category:

- **True Positives (TP):** Correctly predicted positive cases (e.g., correctly identified malignant cases).
- **True Negatives (TN):** Correctly predicted negative cases (e.g., correctly identified benign cases).
- **False Positives (FP):** Mistakenly identified as positive (e.g., benign cases incorrectly classified as malignant – Type I error).
- **False Negatives (FN):** Mistakenly identified as negative (e.g., malignant cases incorrectly classified as benign – Type II error).

The table displays the counts of true positives, true negatives, false positives, and false negatives, clearly showing the distribution of predictions against actual classes. (Kundu, n.d.) In other words, this table shows how many predictions were right or wrong, making it easy to see the

results. The totals are there to help, and the numbers are used to calculate other important measures.

Metric	Formula	Explanation
Accuracy	(TP + TN) / (TP + TN + FP + FN)	Overall correctness of the model
Precision	TP / (TP + FP)	Proportion of positive predictions that were actually correct
Recall	TP / (TP + FN)	Proportion of actual positive cases that were correctly identified by the model
Specificity	TN / (TN + FP)	Proportion of actual negative cases that were correctly identified by the model
F1-score	2 * (Precision * Recall) / (Precision + Recall)	Harmonic mean of precision and recall, providing a balanced measure of model performance

In simple terms for the table chart above, accuracy is like looking at all your answers on a test and seeing how many you got right out of all the questions. Precision is about looking at only the questions you said you got right and figuring out how many you actually got right. Recall is about looking at all the questions that were actually right and seeing how many of those you got right. Specificity is about looking at all the questions that were actually wrong and seeing how many of those you got right. F1-score is like combining precision and recall into one number to get a balanced view of how well you did on the test overall. (Kundu, n.d.)

Conclusion

Our project aimed to identify histologically confirmed skin cancer cases from photos, inspired by the ISIC skin cancer detection competition on Kaggle (Kurtansky, Rotemberg, Gillis, Kose, Reade, Chow). While the competition has concluded, it provided valuable guidance for structuring our project and benchmarking our results against similar initiatives.

The goal of achieving a true positive rate (TPR) of over 80% was not fully met, as our model's sensitivity (recall) for malignant cases reached 36.8%. However, our approach demonstrated significant progress given the challenges of extreme class imbalance, low-resolution images, and resource limitations. Our work validated the foundational architecture and highlighted effective techniques such as class weighting, transfer learning with ResNet50, and tailored preprocessing methods like CLAHE.

Although our model does not currently meet the standards required for clinical application, the results underscore its potential with further refinement. Increased access to higher-quality datasets, improved computational resources, and advanced methods like Google’s LAuReL could enhance model performance, making it suitable for practical use. In conclusion, while we did not fully achieve the >80% TPR goal, this project successfully addressed key challenges in smartphone-based skin cancer detection and laid a solid foundation for future research and development in this critical healthcare area.

References

Aguas, K. C. (2021, December 15). A guide to transfer learning with Keras using ResNet50. *Medium*. Retrieved from https://medium.com

American Cancer Society. (2024). *Cancer Facts & Figures 2024*. Retrieved from https://www.cancer.org/content/dam/cancer-org/research/cancer-facts-and-statistics/annual-cancer-facts-and-figures/2024/2024-cancer-facts-and-figures-acs.pdf

Ansari, S. (2024, November 17). Google AI Introduces LAuReL (Learned Augmented Residual Layer): Revolutionizing Neural Networks with Enhanced Residual Connections for Efficient Model Performance. Retrieved from https://www.marktechpost.com/2024/11/16/google-ai-introduces-laurel-learned-augmented-residual-layer-revolutionizing-neural-networks-with-enhanced-residual-connections-for-efficient-model-performance/

Habif, T. P. (1996). *Clinical Dermatology: A Color Guide to Diagnosis and Therapy*.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern* [1] *recognition* (pp. 770-778).

Pre Trained Model Definition | ENCORD. (n.d.). Retrieved from https://encord.com/glossary/pre-trained-model-definition/#:~:text=A%20pre%2Dtrained%20model%20is,tuned%20for%20a%20specific%20task.&text=Pre%2Dtrained%20models%20are%20often%20trained%20on%20large%2C%20diverse%20datasets,range%20of%20patterns%20and%20features.

Kränke, T., Tripolt-Droschl, K., Röd, L., Hofmann-Wellenhof, R., Koppitz, M., & Tripolt, M. (2023, February 15). *New Ai-algorithms on smartphones to detect skin cancer in a clinical setting-A validation study*. PloS one. https://pmc.ncbi.nlm.nih.gov/articles/PMC9931135/

Kundu, R. (n.d.). Confusion Matrix: How To Use It & Interpret Results [Examples]. V7. https://www.v7labs.com/blog/confusion-matrix-guide

Nicholas Kurtansky, Veronica Rotemberg, Maura Gillis, Kivanc Kose, Walter Reade, Ashley Chow. (2024). ISIC 2024 - Skin Cancer Detection with 3D-TBP. Kaggle. https://kaggle.com/competitions/isic-2024-challenge

Skin cancer. (n.d.). Retrieved from https://www.aad.org/media/stats-skin-cancer

The Skin Cancer Foundation. (2024, February 6). Skin Cancer Facts & Statistics - The Skin Cancer Foundation. Retrieved from https://www.skincancer.org/skin-cancer-information/skin-cancer-facts/%C2%A0

Tschandl, P. (2018). The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions [Dataset]. *Harvard Dataverse*. https://doi.org/10.7910/dvn/dbw86t

Wasike, B. (2023, May 26). How to Build a Deep learning model with Keras and ResNet-50. *Medium*. Retrieved from https://medium.com

Yuan, Geng, et al. "Layer Freezing & Data Sieving: Missing Pieces of a Generic Framework for Sparse Training." *ArXiv (Cornell University)*, 1 Jan. 2022, https://doi.org/10.48550/arxiv.2209.11204. Accessed 21 Apr. 2024.

References Required for Data Licensing

International Skin Imaging Collaboration. SLICE-3D 2024 Challenge Dataset. International Skin Imaging Collaboration https://doi.org/10.34970/2024-slice-3d (2024).

International Skin Imaging Collaboration. SLICE-3D 2024 Permissive Challenge Dataset. International Skin Imaging Collaboration https://doi.org/10.34970/2024-slice-3d-permissive (2024).

The dataset was generated by the International Skin Imaging Collaboration (ISIC) and images are from the following sources: Memorial Sloan Kettering Cancer Center, FNQH Cairns, The University of Queensland, Melanoma Institute Australia, and University of Athens Medical School.