Skinterest Tech
Multi-Modal Skin Condition Classification

AI Studio Project - Skinterest Tech 2A

August-December 2025

BREAK THROUGH TECH

Meet Our Team - Skinterest Tech 2A

Nivi Munjal
University of Maryland,
College Park

Mahek Patel
University of South Florida

Sarah Shafiq
Fordham University

Adhuresa Ukaj
Fordham University

Our AI Studio Coach & Challenge Advisors

Nandini Proothi
AI Studio Coach

Ashley Abid
Challenge Advisor - Skinterest Tech

Thandiwe-Kesi “Thandi” Robins
Challenge Advisor - Skinterest Tech

Significance of Project

“47% of dermatologists felt that their training was inadequate to diagnose skin disease in SOC (Skin of Color) patients.”

Source: Narla, S., Heath, C. R., Alexis, A., & Silverberg, J. I. (2022). Racial disparities in dermatology. Archives of Dermatological Research, 315(5). https://doi.org/10.1007/s00403-022-02507-z

https://pmc.ncbi.nlm.nih.gov/articles/PMC9743121/

SKINTEREST

AI Studio Organization - Skinterest Tech Goals

Mission: Transform the dermatology industry with inclusive and comprehensive care.

- Combine user-reported insights and objective measurements to deliver in-depth analyses and actionable guidance to improve skincare routines.

BREAK THROUGH TECH

Project Purpose

Develop models for effectively identifying skin conditions from diverse user-submitted photos.

Impact

- **Empower** underrepresented communities through accurate skin condition assessments.

- Advance **AI fairness and inclusivity** in dermatology by improving diagnostic accuracy across **diverse skin tones**.

- Showcase the **evolution** of our progress for providing **inclusive skin condition** diagnosis and care through AI solutions.

Our Approach

Image Analysis and Data Exploration of Dataset

September - October 2025

Data Cleaning

September-October 2025

Model Building

November - December 2025

Model Evaluation

November - December 2025

Resources We Leveraged

Google Colab

GitHub

scikit learn

TensorFlow

pandas

Google Drive

Dataset

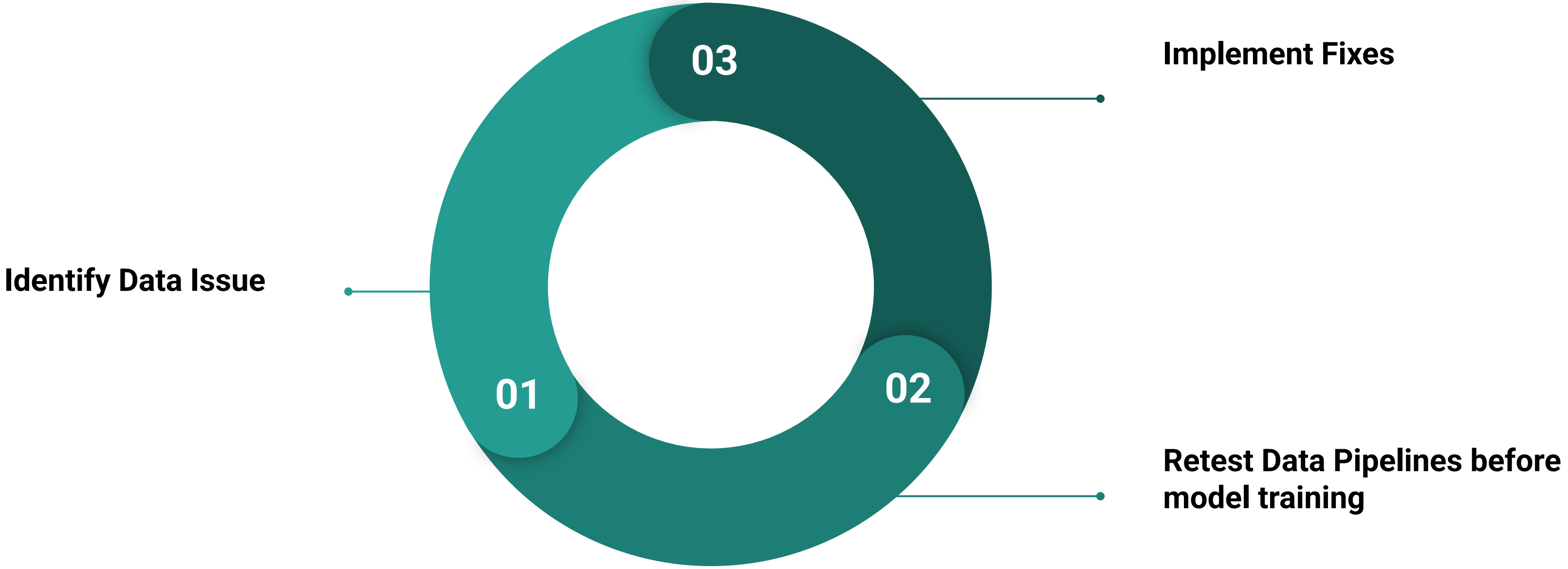
Dataset

Source: Google Skin Condition Image Network (SCIN) open access dataset

The SCIN dataset was crowdsourced from Google Search users to increase the diversity of dermatology images available for public health education and research. It pairs images with detailed self-reported data and expert labels, explicitly focusing on fairness metrics.

Data types: Images, Text, Categorical

Dataset Analysis and Cycle



Dataset Locality

Problem: Dataset too large to be able to save it locally

Current Solution:

- Data remains resident in Google Cloud Storage (GCS)
- Data Analysis and Extraction done though efforts on exporting smaller, summarized result files

Implications:

- Cannot use a simple local pandas pipeline
- Initial data analysis is expected to take longer due to distributed overhead

Handling Text and Images

Approach: Process text/metadata and images differently, using batch workflows and summarized outputs.

Textual Data & Metadata

- Run full analysis on the entire dataset.
- Save condensed statistical outputs as a small CSV/JSON file.
- Download the small file for quick local plotting and reporting.

Image Data

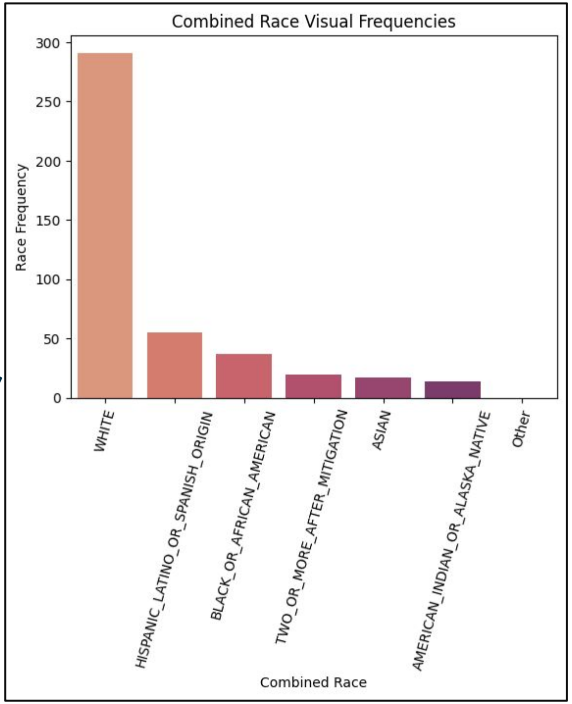
- Process images in batches rather than all at once.
- Download only selected batches for:
  - Manual review/audit
  - Fine-tuning or iterative model updates
- Training jobs read images directly from cloud storage, removing the need for full local copies.

Metadata Data Analysis and Exploration

Data Cleaning and Exploration

Comparison of diversity in old and new skin condition datasets

Old dataset

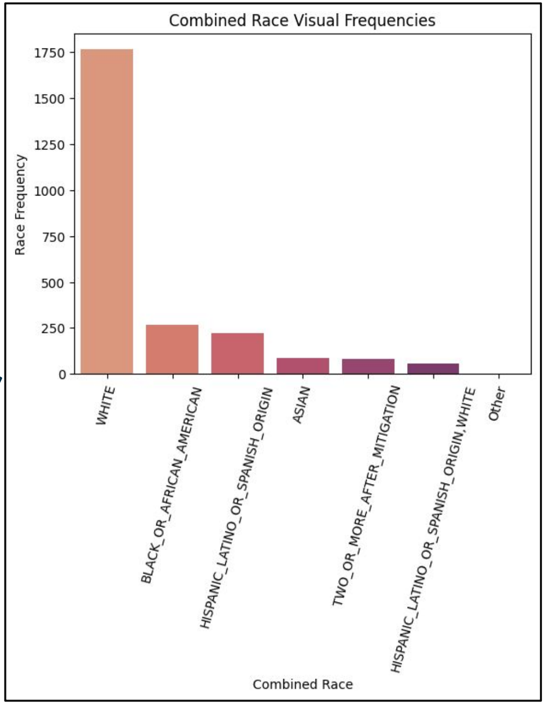


Advantage:

Increased number of diverse skin images

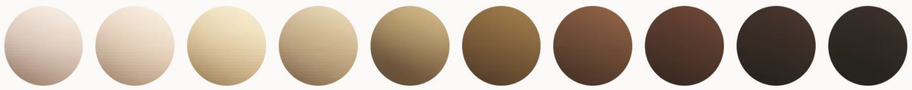
(Observe frequencies of new dataset)

New dataset



Monk Skin Tone (MST) Scale

- Broader spectrum of varying skin tones to represent inclusivity and diversity.
- Developed by Harvard Professor, Dr. Ellis Monk, and partnered with Google to improve Computer Vision (CV) perceptions of skin tone and minimize racial bias in AI/ML applications.
- Variety of skin tones represented in the SCIN dataset



Monk Skin Tone Scale

```
5019
monk_skin_tone_label_india
2.0     2427
3.0     1591
4.0      522
1.0      187
5.0      177
6.0       63
7.0       35
8.0       14
9.0        3
Name: count, dtype: int64
```

```
5005
monk_skin_tone_label_us
2.0      1660
3.0      1265
4.0       687
1.0       577
5.0       361
6.0       248
7.0       137
8.0        57
9.0        11
10.0        2
Name: count, dtype: int64
```

https://skintone.google/

Data Cleaning and Exploration

Challenge: Unlabeled skin condition and confidence score data in new dataset.

```
5033
dermatologist_skin_condition_on_label_name
[]                                                             1972
['Eczema']                                                      120
['Urticaria']                                                    81
['Eczema', 'Allergic Contact Dermatitis']                        67
['Allergic Contact Dermatitis', 'Irritant Contact Dermatitis']   60
['Allergic Contact Dermatitis']                                  42
['Folliculitis']                                                 37
['Urticaria', 'Insect Bite', 'Allergic Contact Dermatitis']      36
['Insect Bite']                                                  27
['Acute dermatitis, NOS']                                        27
['Tinea', 'Psoriasis', 'Eczema']                                 23
['Urticaria', 'Allergic Contact Dermatitis']                     22
['Psoriasis', 'Eczema']                                          21
['O/E - ecchymoses present']                                     20
['Psoriasis']                                                    18
Name: count, dtype: int64
```

```
5033
dermatologist_skin_condition_confidence
[]           1972
[5]           312
[4]           208
[2, 2, 2]     178
[2, 2]        176
[1, 1, 1]     170
[4, 2]        112
[3]           103
[2, 4]        100
[1, 1]         63
[3, 2]         45
[1, 5, 1]      40
[5, 1, 1]      39
[2]            34
[2, 3]         33
Name: count, dtype: int64
```

Data Cleaning and Exploration

Demographics of Unlabeled Data

Challenge: Dropping unlabeled data may unintentionally reduce representation from minority and underrepresented populations, leading to a less diverse dataset.

Possible solution: Using monk scale instead of relying on race column for diversity inclusion

	combined_race	count_unlabeled	percent_unlabeled
0	NaN	1078	54.665314
1	WHITE	658	33.367140
2	BLACK_OR_AFRICAN_AMERICAN	73	3.701826
3	HISPANIC_LATINO_OR_SPANISH_ORIGIN	58	2.941176
4	ASIAN	23	1.166329
5	AMERICAN_INDIAN_OR_ALASKA_NATIVE	17	0.862069
6	TWO_OR_MORE_AFTER_MITIGATION	17	0.862069
7	PREFER_NOT_TO_ANSWER	12	0.608519
8	HISPANIC_LATINO_OR_SPANISH_ORIGIN,WHITE	12	0.608519
9	BLACK_OR_AFRICAN_AMERICAN,WHITE	7	0.354970
10	OTHER_RACE	6	0.304260
11	BLACK_OR_AFRICAN_AMERICAN,HISPANIC_LATINO_OR_S...	4	0.202840
12	NATIVE_HAWAIIAN_OR_PACIFIC_ISLANDER	3	0.152130
13	AMERICAN_INDIAN_OR_ALASKA_NATIVE,WHITE	3	0.152130
14	MIDDLE_EASTERN_OR_NORTH_AFRICAN	1	0.050710

Data Cleaning and Exploration

Challenge: Unlabeled skin condition and confidence score data in dataset.

Many of these images do not have associated race/ethnicity values.

Question: Should we keep this unlabeled data?

- Advantage: Preserve diversity in dataset
- Disadvantage: Images will not have skin condition labels in the model

Final Decision:

- Remove unlabeled data for

  initial supervised learning models.
- Consider hybrid supervised and unsupervised model building in the future

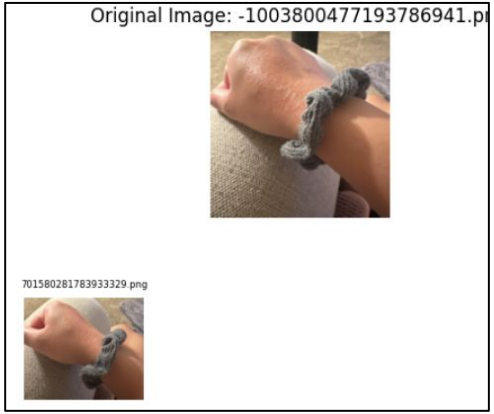
Image Data Analysis and Exploration

Data Cleaning and Exploration

Increased amount of similar images (closeups, varying angles) in new dataset.

Advantage: Captures skin conditions of user-taken skin photos from various zoom ratios, angles, lightning, and more.

```
The total number of closely similar/duplicate image files found is: 598
```



Data Extraction

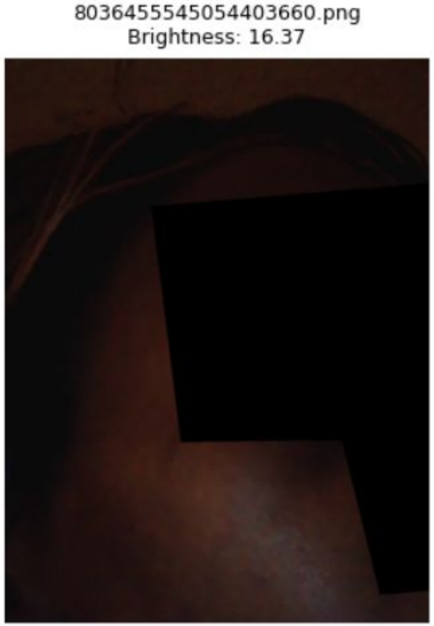
Extracted lightning indicators for model training

image_path	blur	brightness_mean	brightness_std	underexp	overexp	contrast	shadow	sharpness
dataset/images/-1001492676369731180.png	49.8009	125.822289	62.991703	0.152014	0.049442	0.988142	0.162389	49.8009
dataset/images/-1001733364362669777.png	238.390468	74.623044	29.900723	0.03618	0.000554	1	0.03618	238.390468
dataset/images/-1003800477193786941.png	89.631693	128.77948	47.623139	0.005774	0.023419	0.980237	0.055794	89.631693
dataset/images/-1005922060850163675.png	4.17201	102.138025	62.690025	0.179942	0	0.99061	0.010904	4.17201
dataset/images/-1007969568196430462.png	9.460231	147.59777	40.054561	0	0.002477	0.819608	0.107458	9.460231

These features are used to filter low-quality images, evaluate how image quality impacts model performance, and ensure demographic fairness by checking that capture conditions don’t introduce bias.

Images with low brightness such as these that show little to nothing are dropped

Although these images have low brightness, they are not dropped because they show symptoms either through texture or redness. While some other images, just had a dark colored background



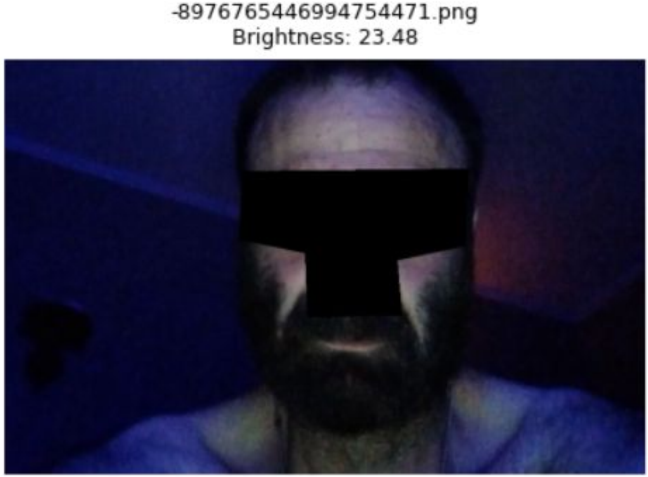
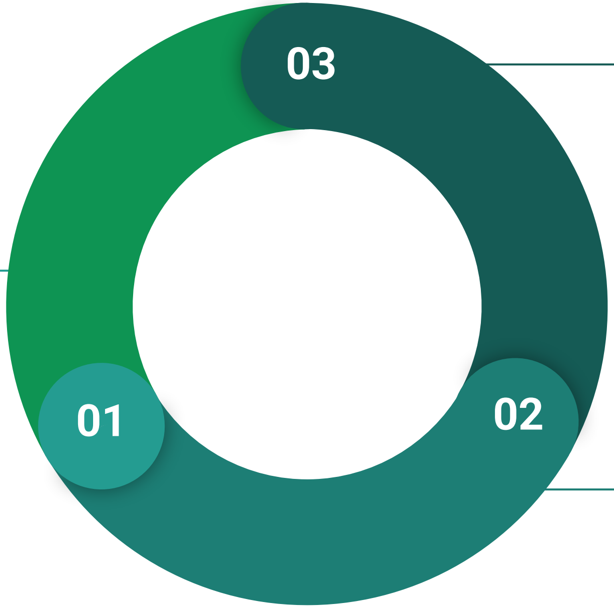


Image Classification Model Building

Model Training Plan



Basic Model Training using CNN, ResNet, Efficient

Train model based on results and evaluation metrics

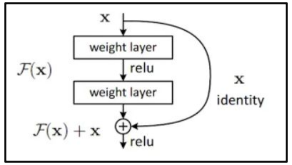
Adding features (aka metadata to our model for training)

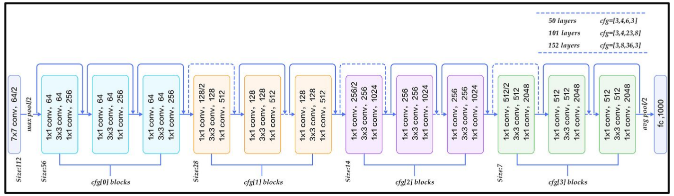
Model Research

CNN Image classification pre-trained model families in consideration: ResNet-50 and EfficientNet

ResNet-50:

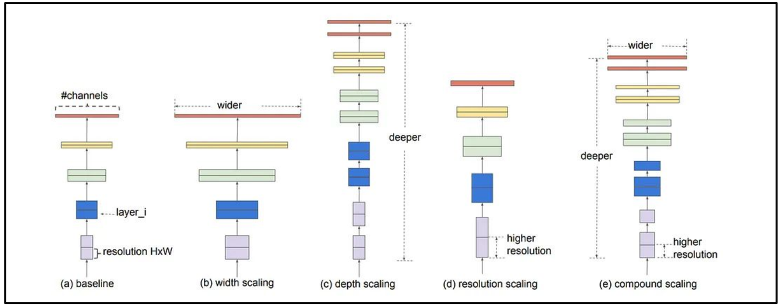
- Emphasis on residual connections.
- Skip layers: Information can skip and go to the following layers.
- Beneficial for training deep networks.





EfficentNet:

- Emphasis on scaling method for depth, width, and resolution, known as Compound Scaling.
- Providers more layers for bigger input images to capture more details/patterns.
- Has been stated to help with efficiency and accuracy.



Sources: [2]

Missing/Deleted files with GCS

- These paths slipped through earlier filtering and caused TensorFlow to fail during training with NotFoundError when reading images.

- Solution:
  - Added a path-validation step using tf.io.gfile.exists() and removed all rows containing invalid image references before splitting the dataset.

Class Imbalance/Rare classes

Some classes had only 1–2 images, which caused:

- Stratified split failures (no samples left for train/test)
- ValueError: train set will be empty
- Required removing or merging extremely small classes

Why This Is a Problem

- Model memorizes rare samples instead of learning real features
- Unstable loss due to unreliable gradients
- Overfitting pattern:
    - Training accuracy spikes
    - Validation accuracy stays low

Image + Metadata Integration

Image + Metadata Model ⟷ Image-Only Dataset

- The multimodal ResNet model required two inputs (image + metadata), but the dataset pipeline only provided images.
- Metadata wasn't correctly aligned with each image, causing input-shape errors, missing metadata, and NaN loss.
- Training couldn't continue until the data pipeline was restructured to deliver both inputs per sample.

Input mismatch → training failed

Modeling Building

CNN: Trained a 2-layer CNN (32/64 filters) fused with a 32-unit metadata branch, followed by a 128-unit Dense layer and softmax output using Adam and 224×224 normalized images

ResNet: Trained a ResNet50 base with a 128-unit Dense classifier and softmax output using Adam on normalized 224×224 images.

Feature groups considered:

- Condition symptoms
- Monk scale tone
- Fitzpatrick
- Race/Ethnicity
- Body location
- Textures

Model Evaluation

Challenges Observed:

- CNN Baseline Overfitting:
  - Training accuracy increased while validation accuracy decreased
  - Memorized training data instead of learning general patterns.

CNN Baseline	Time	Time per step	Accuracy	Loss	Val Accuracy	Val Loss
Epoch 8/10	46s	312ms/step	0.9624	0.2604	0.0809	6.6576
Epoch 9/10	44s	303ms/step	0.977	0.1619	0.099	7.2722
Epoch 10/10	44s	308ms/step	0.9894	0.0927	0.1025	7.6039

- ResNet Baseline stagnant training and validation accuracy
  - Stuck near random chance (~0.15)
  - Indicating issues with class imbalance, input alignment, or insufficient tuning.

ResNet	Time	Time per step	Accuracy	Loss	Val Accuracy	Val Loss
Epoch 8/10	667s	5s/step	0.1572	3.843	0.1607	3.8252
Epoch 9/10	657s	5s/step	0.1606	3.8686	0.1607	3.8332
Epoch 10/10	660s	5s/step	0.1485	3.9116	0.1607	3.8222

Model Iterations

Reflection:

- Identified the need for richer evaluation metrics such as Precision, Recall, F1, Balanced Accuracy, and AUROC to better understand class-specific failures and separability.

Future Work:

- Experiment with learning rate, batch size, dropout/L2 regularization, data augmentation, and unfreezing deeper layers in ResNet for improved learning

Next Steps

1. **Experiment with Additional Feature Combinations**

Continue testing models using different subsets of features, including baseline features, body-part features, and texture-specific features, to evaluate which combinations yield the strongest predictive performance.

2. **Apply Both Supervised and Unsupervised Methods**

Train supervised models to measure classification accuracy, and complement this with unsupervised approaches (e.g., clustering or dimensionality reduction) to uncover hidden patterns and validate feature separability.

3. **Web Application of Skin Condition Classification**

Web application platform where users can submit their own skin condition images. Get real-time skin condition evaluation from the models.

Thank you!

Questions?

Appendix

Model Research Sources

- [1] Hartanto, D., & Herawati, R. (2024). COMPARATIVE ANALYSIS OF EFFICIENTNET AND RESNET MODELS IN THE CLASSIFICATION OF SKIN CANCER. Proxies : Jurnal Informatika, 7(2), 69–84. https://doi.org/10.24167/proxies.v7i2.12468
- [2] Randellini, E. (2023, January 5). Image classification: ResNet vs EfficientNet vs EfficientNet_v2 vs Compact Convolutional…. Medium. https://medium.com/@enrico.randellini/image-classification-resnet-vs-efficientnet-vs-efficientnet-v2-vs-compact-convolutional-c205838bbf49

Dataset

Total Contributions	5,000+	Volunteer contributions collected via a consented image donation application
Total Images	10,000+	Up to 3 images per case (Close-Up, At-Distance, At-An-Angle)
Data Types	Images, Text, Categorical	Includes self-reported data and expert labels

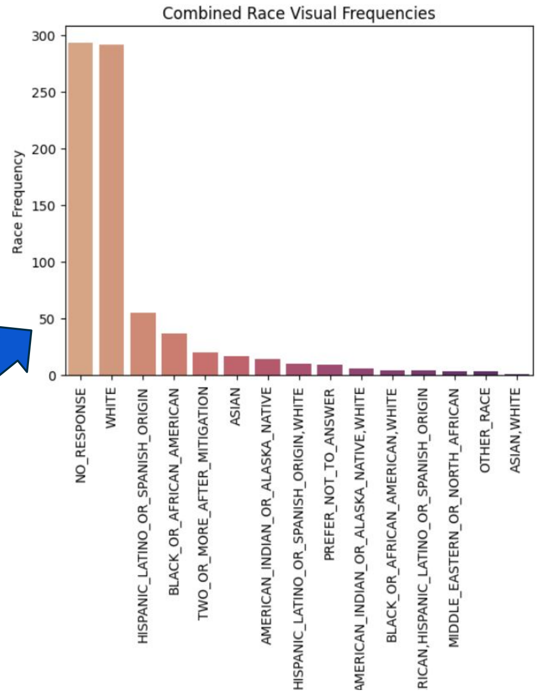
Key attributes and fairness metrics

Expert Labeling	• dermatololgist_labels • dermatologist_confidence
Demographics	• race_ethnicity • sex_at_birth • age_group
Fairness metrics	• monk_skin_tone_label_india/_us
Skin Type	• fitzpatrick_skin_type • dermatologist_fitzpatrick_skin_type_label
Clinical History	• body_parts • Condition_symptoms • conditon_duration

Data Cleaning and Exploration

Comparison of diversity in old and new skin condition datasets

Old dataset



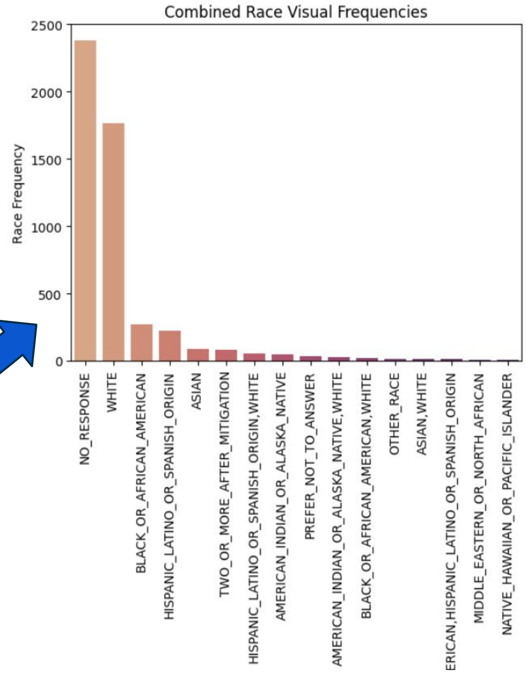
Advantage:

Increased number of diverse skin images

(Observe frequencies of new dataset)

```
5033
combined_race
NO_RESPONSE                                                          2381
WHITE                                                                1762
BLACK_OR_AFRICAN_AMERICAN                                             267
HISPANIC_LATINO_OR_SPANISH_ORIGIN                                     224
ASIAN                                                                  85
TWO_OR_MORE_AFTER_MITIGATION                                           83
HISPANIC_LATINO_OR_SPANISH_ORIGIN,WHITE                                55
AMERICAN_INDIAN_OR_ALASKA_NATIVE                                       48
PREFER_NOT_TO_ANSWER                                                   34
AMERICAN_INDIAN_OR_ALASKA_NATIVE,WHITE                                 25
BLACK_OR_AFRICAN_AMERICAN,WHITE                                        20
OTHER_RACE                                                             16
ASIAN,WHITE                                                            11
BLACK_OR_AFRICAN_AMERICAN,HISPANIC_LATINO_OR_SPANISH_ORIGIN            11
MIDDLE_EASTERN_OR_NORTH_AFRICAN                                         7
NATIVE_HAWAIIAN_OR_PACIFIC_ISLANDER                                     4
Name: count, dtype: int64
```

New dataset



Data Cleaning and Exploration

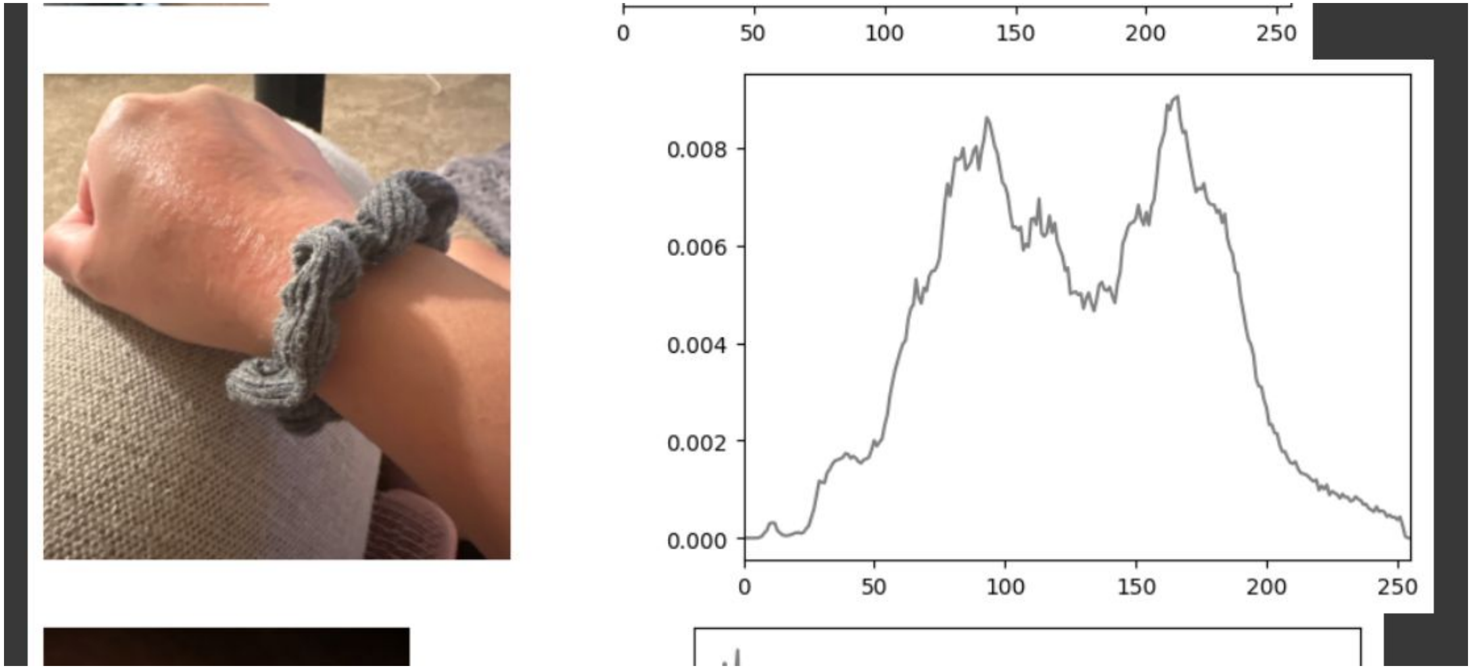
Demographics of unlabeled data

Challenge: Dropping unlabeled data may unintentionally reduce representation from minority and underrepresented populations, leading to a less diverse dataset.

	count	percent
race_ethnicity_white	680.0	34.482759
race_ethnicity_black_or_african_american	84.0	4.259635
race_ethnicity_hispanic_latino_or_spanish_origin	74.0	3.752535
race_ethnicity_asian	23.0	1.166329
race_ethnicity_american_indian_or_alaska_native	20.0	1.014199
race_ethnicity_prefer_not_to_answer	12.0	0.608519
race_ethnicity_other_race	6.0	0.304260
race_ethnicity_native_hawaiian_or_pacific_islander	3.0	0.152130
race_ethnicity_middle_eastern_or_north_african	1.0	0.050710

Possible solution: Using monk scale instead of relying on race column for diversity inclusion

Extracted brightness histograms so we can train model on images with higher brightness/texture



Create global functions for easier access to data

Variables

- df_original: The full, unmodified dataset containing all cases, metadata, and image paths. This is equivalent to Globals.cases_and_labels_df and can be used like a normal pandas DataFrame.
- df_filtered: A working copy of the dataset that you can safely modify, filter, or clean without affecting the original.
- image_dir : Use this directory to access images within google cloud

Functions

- read_image_from_gcs(gcs_path)
  - Downloads and decodes an image directly from your GCS bucket using the path stored in the dataset (e.g. "dataset/images/12345.png").
- get_all_image_paths()
  - Extracts all unique image paths from the three image columns (image_1_path, image_2_path, image_3_path) in the dataset.
- show_case_images(case_id)
  - Displays all available images for a given case_id directly from GCS.
- convert_to_binary_var(col_name)
  - convert to binary values

Note: add instructions if we want to analyze/change/decode all images together

Data Cleaning and Exploration

```
5033
related_category
RASH                        2876
NO_RESPONSE                 1254
OTHER_ISSUE_DESCRIPTION      414
LOOKS_HEALTHY                290
ACNE                          74
GROWTH_OR_MOLE                45
PIGMENTARY_PROBLEM            37
NAIL_PROBLEM                  20
OTHER_HAIR_PROBLEM            12
HAIR_LOSS                     11
Name: count, dtype: int64
```

Challenge: Unlabeled skin condition and confidence score data in dataset.

Many of these images do not have associated race/ethnicity values.

Question: Should we keep this unlabeled data?

- Advantage: Preserve diversity in dataset
- Disadvantage: Images will not have skin condition labels in the model

Additionally, if the unlabeled images contain skin conditions, what should we use for images of skin without conditions for model training?

Option: related_category column contains “LOOKS_HEALTHY” value.

Model Research

CNN Image classification model families in consideration: ResNet-50 and EfficientNet

- Pre-trained CNN models for image classification
- Previous research and usage of these models for skin cancer image detection/classification
- Study comparing the two models for Skin Cancer Image Classification:

Table 1. Accuracy Table Flatten – Drop 0.2

Model	Train	Validation	Test
ResNet50	94.56%	78.56%	80.72%
EfficientNet B0	98.73%	86.43%	84.12%
EfficientNet B1	98.36%	85.73%	86.41%
EfficientNet B2	99.04%	87.92%	87.01%
EfficientNet B3	97.64%	85.53%	84.62%
EfficientNet B4	98.98%	84.93%	85.01%
EfficientNet B5	98.03%	85.93%	84.52%
EfficientNet B6	99.45%	85.93%	85.51%
EfficientNet B7	98.94%	85.93%	86.91%

The study's observation:

EfficientNet resulted in higher validation and test accuracy for skin cancer image classification throughout their trials.

Sources: [1]

Model Research

Overall Performance in Accuracy for CNN Models

Observation: EfficientNet Models outperform

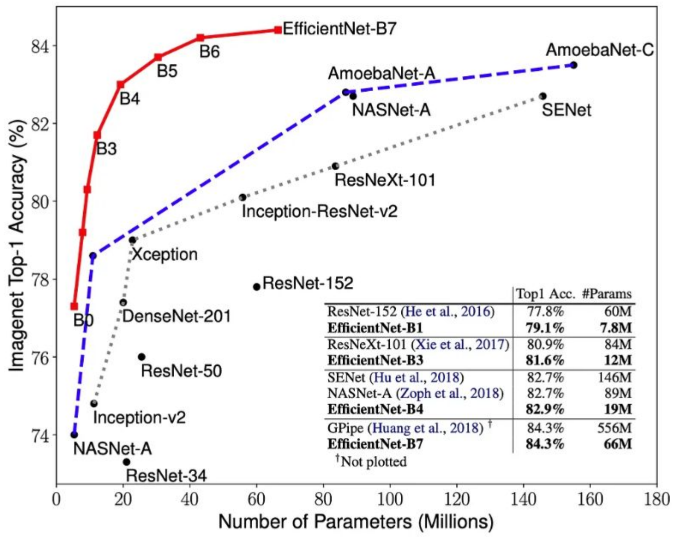


Figure 6. Model Size vs. ImageNet Accuracy from [2]

Sources: [2]