# Quantifying Cutaneous Dermatomyositis: A Novel Image-based Approach

**Prakruthi Pradeep**

CMU-CS-24-157

December 2024

Computer Science Department
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

**Thesis Committee:**

Artur W. Dubrawski (Chair)
Bhiksha Raj

*Submitted in partial fulfillment of the requirements for the degree of Master of Science in Computer Science.*

**Keywords:** Disease Severity Prediction, Image Analysis, Handcrafted Feature Extraction, Semantic Image Segmentation, Image Classification, 3D Imaging, Telemedicine Application-based Imaging, K-Means Clustering, Grad-CAM Visualization, Cutaneous Dermatomyositis, CDM, CDASI

# Abstract

Dermatomyositis (DM) is a rare autoimmune disease characterized by chronic muscle inflammation, weakness, and skin rashes. Cutaneous Dermatomyositis (CDM), the skin manifestation of the disease, typically presents as purple or red rashes on the eyelids, joints, knuckles, and other areas; while there is no cure, treatment can alleviate symptoms, and monitoring disease progression is crucial. This study introduces a novel image-based approach for assessing CDM severity, aiming to create an objective, predictive model based on dermatological images, with expert assessments of the Cutaneous Dermatomyositis Activity Score as the gold standard. Through our collaboration with clinicians at the University of Pittsburgh Medical Center, we analyze a dataset of high-resolution 3D in-clinic hand images from DM patients. Key clinical features, including the extent, intensity and texture of the rash, are analyzed alongside CNN-based image features, enabling a comprehensive assessment of disease severity. We evaluate multiple state-of-the-art image classification models, fine-tuning them on our dataset to optimize performance. Our approach includes utilizing semantic image segmentation to accurately highlight regions of interest, with significant improvements achieved through this integration. Our study lays the groundwork for the use of patient-taken images for remote monitoring, demonstrating the potential for patients to track their condition at home. By combining clinical insights with advanced image analysis, this work contributes to improved automated assessment of CDM and better monitoring of disease progression.

## Acknowledgments

I would like to express my heartfelt gratitude to my research advisor Professor Artur W. Dubrawski, Dr. Nantakarn Pongtarakulpanit, and Dr. Rohit Aggarwal, for their invaluable guidance and advice throughout this journey. I would also like to thank my parents, sister, and friends, whose unconditional love and support have been a source of strength and comfort for me.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

Figure 1.1: An image of Cutaneous Dermatomyositis rash on the hands. The rash often appears patchy, with purple or red discolorations, and develops on muscles used to extend joints such as the knuckles. The rashes typically distribute symmetrically across both hands.

Dermatomyositis (DM) is a rare multi-system autoimmune disease that can involve chronic muscle inflammation, muscle weakness, and skin rash. Cutaneous Dermatomyositis (CDM) is the skin manifestation of Dermatomyositis, with the rash often appearing patchy, with purple or red discolorations, and characteristically developing on the eyelids and on the extensor surface of the joints, including knuckles, elbows, knees, and toes. Rashes may also occur on the face, neck, upper chest, back, and other locations, with potential swelling in the affected areas [34]. The disease is quite rare, with an estimated fewer than 5000 people in the United States being affected [35], and can affect adults and children, occurring in the late 40s to early 60s in adults and appearing between 5 and 15 years of age in children. Additionally, women are twice as likely as men to be diagnosed with the disease [31] [8]. The rash can sometimes occur without obvious muscle involvement and often becomes more evident with sun exposure. Periods of symptom improvement can occur with medical therapy and skin treatment, with sunscreen or antihistamine drugs potentially helping clear the skin rash [31].

This work presents a novel image-based approach to assessing CDM severity in patients, with several contributions to the fields of medical image analysis and classification. Our main goal is the development of a predictive model aimed at generating an objective image-based disease

severity score, with an expert assessment of the Cutaneous Dermatomyositis Disease Area and Severity Index (CDASI) score [53] serving as the gold standard. Automating the severity evaluation of a CDM rash will assist clinicians in tracking the progression of the disease over time for improvement after initiating treatment.

This study is made possible through our collaboration with clinicians from the Division of Rheumatology and Clinical Immunology at the University of Pittsburgh Medical Center. This collaboration gave us access to valuable real-world data and a more comprehensive understanding of the clinical features most indicative of CDM severity. Our first step was to analyze these key clinical features such as the extent, intensity, and texture of the rash. These features are initially extracted using K-means clustering, an unsupervised learning technique that groups pixels with similar characteristics, and their predictive power is evaluated using three robust classification algorithms. We observed that textural features played a significant role in improving our classification algorithm's ability to classify the rash severity; Specifically, the features of contrast and dissimilarity helped in identifying heterogeneous regions and capturing textural irregularities.

One of the challenges, however, in assessing the rash is accurately highlighting the region of interest. Semantic image segmentation helps us achieve this objective by delineating the rash regions, ensuring a more accurate evaluation of the extent of the disease. We fine-tuned multiple state-of-the-art semantic segmentation models on our dataset, to produce segmented crops that can be fed into our clustering algorithms for subsequent feature extraction. This pre-processing step aims to improve the classifier's ability to differentiate between healthy and affected skin, leading to a more accurate measurement of disease severity.

We then investigated the potential of Convolutional Neural Networks (CNNs) to extract complex patterns and hierarchical features from our image data, and automatically identify and quantify the characteristics of the rash. After thorough testing, we determined that combining CNN-based feature extraction with the Support Vector Machine (SVM) classifier yielded the best performance, achieving an accuracy of approximately $85\%$, which the clinicians deemed to be within a satisfactory range. This approach was particularly effective at distinguishing between very mild and severe cases of CDM, which is crucial for clinicians. We also assessed the effectiveness of fine-tuning pre-trained CNNs on our dataset and evaluated these results against our previous pipelines involving explicit feature extraction and classification.

Finally, we utilized Grad-CAM [47] to generate class activation maps to visualize the regions of focus for our fine-tuned model. This provided the clinicians with a degree of transparency and explainability into the otherwise black-box CNN, to better understand and interpret which parts of the hand images the model is focusing on when making a prediction.

Ultimately, this study leveraged key clinical insights provided by domain experts, along with the more complex, hierarchical features produced by deep learning models, to create a framework for an efficient, automated assessment of CDM. In summary, our work presents the following contributions:

- We partnered with clinicians at UPMC to curate a novel dataset of in-clinic and smartphone-based Cutaneous Dermatomyositis hand images
- We leveraged the K-means clustering algorithm to extract key clinical features including rash extent, intensity and texture, and further refined this feature extraction process with

the use of region of interest segmentation as a pre-processing step

- We improved on the performance of the handcrafted features with CNN-extracted features, achieving a top accuracy of 85% with the SVM classifier
- We applied the Grad-CAM visualization technique to provide clinicians with visual insights into our fine-tuned model's decision-making process, enhancing interpretability and trust

# Chapter 2

# Related Work

## 2.1 Image Segmentation

Image segmentation is the process of partitioning an image into meaningful regions corresponding to areas of interest [25]; Medical image segmentation is typically utilized to assist clinicians in diagnosing, planning treatments, or monitoring disease progression. Historically, segmentation in medical imaging involved thresholding and region-growing techniques, where simple intensity-based methods were applied to delineate object boundaries. The growth of imaging technology and machine learning led to more sophisticated methods like edge detection and active contours, as well as statistical methods, such as K-means clustering and support vector machines. However, it was the deep learning that revolutionized segmentation, which became much more automated and accurate, with CNNs drastically improving the performance of segmentation tasks by learning features directly from data [43].

A notable deep learning-based architecture for image segmentation is the U-Net architecture [41], which was specifically designed for medical semantic segmentation and has become a gold standard in the field. Semantic segmentation associates a label or category with every pixel in an image, enabling accurate localization of areas of interest [21]. The U-Net is a fully convolutional network with an encoder-decoder structure, where the encoder extracts features while the decoder rebuilds the segmented output.

The U-Net is particularly effective because of its ability to work with relatively small datasets, which is a common issue in medical imaging, where labeled data is scarce. It is especially effective for tasks like segmenting tumors, organs, and other medical structures, and helps clinicians by automating the delineation of critical structures, reducing the time spent on manual annotation.

A typical encoder-decoder image segmentation model, as seen in Figure 2.1 follows an architecture that is designed to effectively capture spatial features in images while preserving the fine details necessary for accurate segmentation. The encoder part of the network consists of convolutional layers followed by max pooling layers. These layers extract hierarchical features from the image, progressively reducing the spatial dimensions while increasing the number of feature maps. The encoder captures low-level features such as edges, textures, and shapes, which are important to distinguish between different structures. The bottleneck represents the deepest

part of the network, where the model captures the most abstract features. Afterwards, the decoder consists of up-sampling layers and skip connections from the corresponding layers in the encoder. These skip connections help retain fine-grained spatial details that are lost during down-sampling in the encoder. The decoder progressively increases the spatial resolution of the feature maps, reconstructing the segmentation mask in the process. Finally, the final layer typically uses a softmax activation function to output a pixel-wise probability map representing the probability or likelihood that each pixel belongs to a certain class [6].

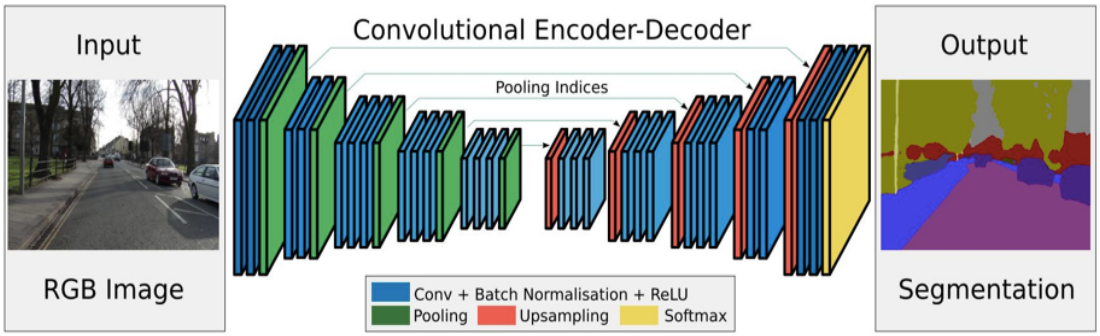


Figure 2.1: Typical Encoder-Decoder architecture. The encoder consists of convolutional layers followed by max pooling layers and the decoder consists of up-sampling layers. The final layer uses a softmax activation function.

## 2.2 Image Classification

Image classification involves classifying images into different categories based on the various visual features present in the image. This is critical in medical imaging, where classifying the severity of a disease can potentially guide clinical decisions and treatment plans. Historically, severity classification in medical imaging was done manually by clinicians, but the advent of deep learning has automated this process, making it more efficient and reliable. The typical workflow for severity classification involves using CNNs to learn relevant patterns in images and classify them into severity levels. Transfer learning approaches are frequently used to overcome the challenge of limited annotated data, leveraging pre-trained models on large datasets and fine-tuning them for medical tasks [42].

A typical image classification model, as seen in Figure 2.2, starts with a pre-trained convolutional base, such as a ResNet, which extracts features from the input image. These pre-trained networks are well-suited for feature extraction, as they have already learned to detect low-level and high-level features from large image datasets. The feature extractor is followed by one or more fully connected layers that classify the extracted features into different classes or output a continuous score. For multi-class classification, cross-entropy loss with softmax activation in the final layer is typically used [32].

As mentioned earlier, transfer learning is often applied in medical image classification as it allows the model to benefit from the vast knowledge learned on large, non-medical datasets and

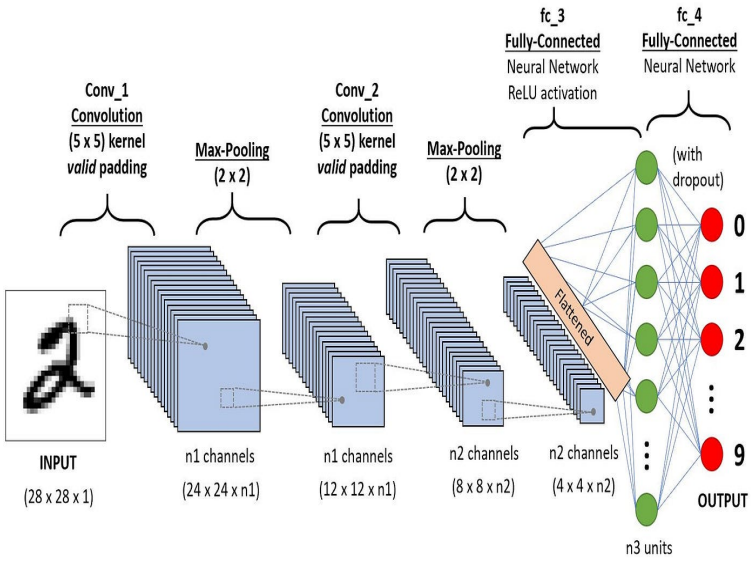


Figure 2.2: Typical CNN architecture. The architecture consists of convolutional and pooling layers, which form the pre-trained convolutional base, followed by fully connected layers that classify the extracted features.

adapt it to the specific medical domain which often has relatively small datasets. For severity classification, fine-tuning the pre-trained network ensures that the model can adjust its weights to better classify specific features relevant to the severity of the disease in question. Deep learning techniques such as these have been increasingly applied to the study of skin rashes, with the growing demand for accurate and automated diagnostic tools. Some of the most common skin diseases associated with rashes that have been extensively studied in the context of machine learning include Psoriasis, Eczema, Contact Dermatitis and Rosacea.

## 2.3 Segmentation and Classification of Skin Disease Images

Cutaneous Dermatomyositis is a relatively under-researched condition, and we therefore sought out prior work on diseases that manifested similarly to CDM, as red rashes or textured skin legions. Prior work on medical image segmentation for diseases similar to CDM include Rahman Attar et al. (2023) [40], who presented a fully automated method for assessing eczema severity using digital camera images. The authors developed a model, EczemaNet2, to detect eczema lesions from 1,345 dermatological images using data augmentation and pixel-level segmentation. They concluded that the quality and robustness of eczema lesion detection increased by approximately 25% and 40% with the use of pixel-level segmentation, with no real impact on the performance of the downstream severity prediction, however.

We aim to integrate a variety of methods including segmentation, feature extraction, and classification techniques to create a comprehensive solution for accurate disease severity prediction. Ahammed et al. (2022) [1] presented an innovative approach for automated skin disease

detection and lesion segmentation utilizing some of these methods. The authors introduced the GrabCut algorithm, enhanced by K-means clustering and the HSV color space, for automatic lesion segmentation. Key features of the lesions are then extracted through the use of the Gray Level Co-occurrence Matrix (GLCM). The paper finally investigates the performance of various machine learning algorithms including SVM, KNN and Decision Tree, for skin disease classification on the ISIC 2019 and HAM 10000 datasets.

We also aim to incorporate self-training algorithms and class-activation maps in future work to augment our current analysis. Wang et al. (2023) [51] propose a Collaborative Learning Deep Convolutional Neural Networks (CL-DCNN) model, based on the teacher-student learning method for dermatological segmentation and classification. Their method leverages self-training to generate high-quality pseudo-labels, which are screened and selectively retrained via the classification network, enhancing the segmentation quality. Additionally, the model incorporates class activation maps to refine the segmentation network's accuracy and utilizes lesion segmentation masks to aid the classification network in better recognizing skin diseases.

All of these studies present innovative and effective techniques for skin legion segmentation and classification. We aim to incorporate some of the ideas discussed in this section in our analysis. Cutaneous Dermatomyositis is not a disease that has been extensively studied in the past and therefore, our work provides a foundation for future research into the disease.

# Chapter 3

# Cutaneous Dermatomyositis Dataset

## 3.1 DART Study

Clinicians from the Division of Rheumatology and Clinical Immunology at the University of Pittsburgh Medical Center investigated the feasibility of telemedicine in evaluating CDM skin rashes compared to traditional in-clinic assessments. To this end, they also aimed to assess the validity, reliability, and responsiveness of the three new image-based CDM skin rash assessments, consisting of in-clinic 3D imaging, telemedicine application-based imaging, and patient rash mapping. CDM patients, according to the 2017 EULAR/ACR criteria, were prospectively enrolled in an observational study called DART or Dermatomyositis Assessment of Rash via Telemedicine. Each patient underwent evaluations by two independent rheumatologists ($MD_1$ and $MD_2$) during both in-clinic and telemedicine visits, which occurred 2-4 weeks apart. The Cutaneous Dermatomyositis Disease Area and Severity Index (CDASI) [50] were scored during these evaluations.

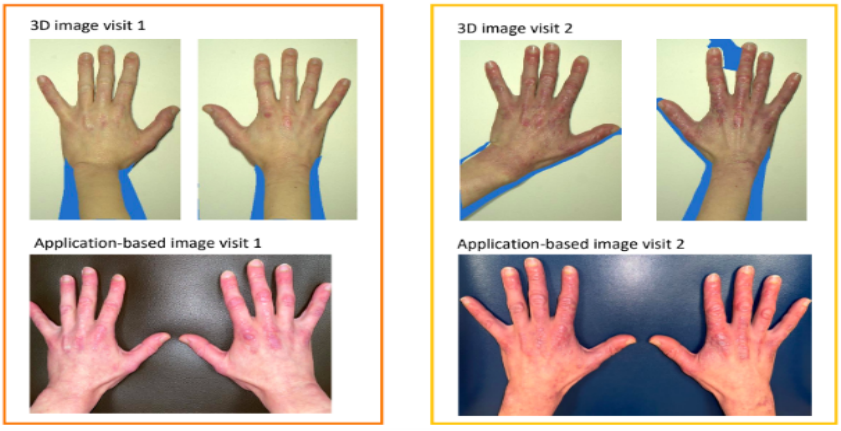


Figure 3.1: Paired left and right-hand photographs of in-clinic 3D images (top) and smartphone images (bottom). For both types of visits, there were approximately 2 visits per patient and 1 set of images per visit.

As a preliminary result, a total of 27 DM patients underwent evaluation. 26 patients were Caucasian-American, and 1 patient was African-American, with a majority of the patients being female, approximately 82.6%. The patients had a mean age of 48.6 ± 17.4 years and a median disease duration of 38.0 months. For both types of visits, there were approximately 2 visits per patient and 1 set of images per visit, capturing skin rashes on various body areas, including the hands, the upper chest, the upper back, and the face. This resulted in a total of approximately 46 sets of 3D in-clinic images being captured. For this research study, the focus will be specifically on the in-clinic hand images, with our initial dataset consisting of 90 hand (left and right) images, as shown in Figure 3.1 To simplify our analysis, these 3D images were converted into 2D representations that were used throughout this study.

It is important to note that the rashes presented differently in the African-American patient, with the rash area being difficult to distinguish and map for the clinicians. Therefore, after consulting with the clinicians, the images associated with this patient were removed from our dataset. The lack of diversity in skin color in our dataset is a limitation that we hope to overcome in the future, and we aim to address the challenges posed by darker skin tones by utilizing image-enhancing techniques to increase rash visibility and incorporating non-chromatic features such as texture.

Figure 3.2: The Vectra software application. This is utilized by the clinicians to map and measure the extent and intensity features of the rash for the 3D in-clinic images

The 3D in-clinic images were taken by a research coordinator using a VECTRA H1 camera, enhanced by studio-quality lighting to ensure optimal visualization of skin topography. The VECTRA H1 camera is a handheld imaging system that provides clinical-quality high-resolution 3D imaging, using different lenses and filters to give a complex, in-depth view of the skin [44]. The VECTRA application software, showcased in Figure 3.2, is utilized by clinicians to study and analyze the in-clinic images, enabling them to map and measure features such as the rash extent and intensity [44]. The telemedicine images were taken by patients during visits using the SkinIO smartphone application [48]. The quality of the telemedicine images is generally lower than that of the in-clinic data, with noticeable variations in lighting and angle. These

discrepancies should be addressed when applying the image analysis techniques used for the in-clinic images to the telemedicine dataset. This will likely require the introduction of pre-processing steps to normalize factors such as lighting and hand positioning across the images.

## 3.2 CDASI Scoring

| STUDY ID | | STUDY VISIT | | VISIT DATE | |
|---|---|---|---|---|---|

**Cutaneous Dermatomyositis Disease Area and Severity Index (CDASI) *ver02***

Select the score in each anatomical location that describes the most severely affected dermatomyositis-associated skin lesion

| | Activity | | | Damage | | |
|---|---|---|---|---|---|---|
| Anatomical Location | Erythema | Scale | Erosion/ Ulceration | Poikiloderma (Dyspigmentation or Telangiectasia ) | Calcinosis | Anatomical Location |
| | 0-absent 1-pink; faint erythema 2-red 3-dark red | 0-absent 1-scale 2-crust; lichenification | 0-absent 1-present | 0-absent 1-present | 0-absent 1-present | |
| Scalp | | | | | | Scalp |
| Malar Area | | | | | | Malar Area |
| Periorbital | | | | | | Periorbital |
| Rest of the face | | | | | | Rest of the face |
| V-area neck (frontal) | | | | | | V-area neck (frontal) |
| Posterior Neck | | | | | | Posterior Neck |
| Upper Back & Shoulders | | | | | | Upper Back & Shoulders |
| Rest of Back & Buttocks | | | | | | Rest of Back & Buttocks |
| Abdomen | | | | | | Abdomen |
| Lateral Upper Thigh | | | | | | Lateral Upper Thigh |
| Rest of Leg & Feet | | | | | | Rest of Leg & Feet |
| Arm | | | | | | Arm |
| Mechanic's Hand | | | | | | Mechanic's Hand |
| Dorsum of Hands (not over joints) | | | | | | Dorsum of Hands (not over joints) |
| Gottron's-Not on Hands | | | | | | Gottron's-Not on Hands |

Extent

**Gottron's -Hands**

| Examine patient's hands and double score if papules are present | | Ulceration | Examine patient's hands and score if damage is present | |
|---|---|---|---|---|
| 0-absent 1-pink; faint erythema 2-red erythema 3-dark red | | | 0-absent 1-dyspigmentation 2-scarring | |

**Periungual**

| Periungual changes (examine) | | | |
|---|---|---|---|
| 0-absent 1-pink; red erythema/microscopic telangiectasias 2-visible telangiectasias | | | |

**Alopecia**

| Recent Hair loss (within last 30 days as reported by patient) | | | |
|---|---|---|---|
| 0-absent 1-present | | | |
| (For the activity score, please add up the scores of the left side, i.e. Erythema, Scale, Excoriation, Ulceration, Gottron's, Periungual, Alopecia) | Total Activity Score _____ | (For the damage score, add up the scores of the right side, i.e. Poikiloderma, Calcinosis) | Total Damage Score _____ |

Signature_____ Date: ____/____/_____

Figure 3.3: The CDASI form used by clinicians to assess CDM severity. It is a partially validated, clinician-scored, one-page outcome measure used to assess skin disease in CDM patients, evaluating the skin in 15 anatomic locations and comprising of two separate scores based on activity and damage.

Furthermore, as mentioned earlier, we are provided with MD assessments of the CDASI score for all of the images in our dataset. In the past, the cutaneous manifestations of DM were among the least systematically studied aspects of this disease due to a lack of validated instruments for reliably determining the impact of therapy on CDM disease activity. [50] The CDASI, a partially validated, clinician-scored, one-page outcome measure used to assess skin disease in CDM patients, was developed and validated for use by dermatologists as a reliable measure in direct response to this need. An example of the CDASI form is shown in Figure 3.3. Previous studies have demonstrated that the CDASI has the best responsiveness to clinical change when compared to other outcome measures that assess cutaneous manifestations of DM. [50] It evaluates the skin in 15 anatomic locations and is comprised of two separate scores based on activity and damage, among other factors, with higher scores indicating greater disease severity. Activity indicates the current level of disease activity and damage measures the long-term, irreversible changes caused by CDM. The MDs utilized activity to evaluate severity, specifically the three

factors of erythema, scale, and erosion. [50][3] The CDASI scores for the hand images range from 0 to 14, with lower scores indicating milder severity and higher scores reflecting greater severity. Images of the right and left hands are evaluated simultaneously by one CDASI score, as the rashes almost always distribute symmetrically in patients.

## 3.3 Redness and Area Features

| Spearman's rho | Rash area | Rash area *redness | Expert assessment | Patient assessment |
|---|---|---|---|---|
| Rash area | | 0.897 | 0.772 | 0.648 |
| Rash area *redness | 0.897 | | 0.721 | 0.636 |
| Expert assessment | 0.772 | 0.721 | | 0.814 |
| Patient assessment | 0.648 | 0.636 | 0.814 | |

Figure 3.4: Spearman's Rho between handcrafted features and patient and expert CDASI assessments. The values are consistently above 0.6, indicating a substantial degree of association between rash redness and area and the CDASI scores.

In addition to the CDASI scores, the clinicians provided us with 4 features per image, capturing the extent and intensity of the rash:

- The area of the rashes in $cm^2$
- The area of the entire hand in $cm^2$
- The redness of the rash, measured as the average $a^*$ of the rash pixels in the $L^*a^*b^*$ color space [20]
- The redness of the entire hand, measured as the average $a^*$ of the normal skin pixels in the $L^*a^*b^*$ color space [20]

In the upcoming sections, we will explore the reasoning behind measuring redness in this manner and representing our images in the $L^*a^*b^*$ color space. The clinicians determined that the features of rash redness and area highly correlate with both expert and patient assessments of CDASI. Figure 3.4 showcases Spearman's rho [49], a measure of the strength of association between two variables, between both expert and patient CDASI assessments and the rash area and redness. Spearman's Rho ranges from -1 to 1 with positive values indicating a positive association, where when one variable increases, so does the other. From the table, it can be observed that the Rho values are consistently above 0.6, indicating a substantial degree of positive association between rash redness and area and the CDASI scores. This analysis essentially demonstrates that the larger and redder the rash is, the higher its CDASI score and the more severe the rash.

The area feature was transformed into the ratio of rash area to normal hand area to account for variations in the hand size, and the ratio of redness between the rash and surrounding normal

skin was added as a feature to normalize against skin tone variation. These three variables of relative area, redness, and relative redness, form the initial handcrafted features we use to predict the CDASI score.

## 3.4 Collaboration with Domain Experts

A key feature of this work is the close collaboration with the medical research team at UPMC. This collaboration greatly enhanced the quality of our research and provided us with several advantages. It granted us access to valuable real-world data and fostered a deeper understanding of the clinical features most representative of CDM. Furthermore, it helped establish clear benchmarks for our research, allowing us to determine what would be considered acceptable and valuable from a clinical standpoint, and evaluate how closely aligned our findings are with clinical expectations.

# Chapter 4

# Preliminary Analysis

## 4.1 Interrater Relability

Our preliminary analysis included measuring the degree of agreement between the CDASI ratings of the two MDs, for which we computed a metric of interrater reliability, the weighted Cohen's kappa. Cohen's Kappa is a statistical measure used to quantify the level of agreement between two raters who each classify items into categories [15]. We used the weighted version of Cohen's kappa as it applies to ordinally scaled samples, like severity scores [16].

The weighted Cohen's kappa was found to be: $0.561 \pm 0.0615$, with a confidence interval $CI_{95\%} = [0.441, 0.682]$. Landis and Koch (1977)[27] provided a widely referenced interpretation of Cohen's Kappa to assess the level of agreement between raters. According to their work, a Cohen's Kappa value of 0.41–0.60 indicates moderate agreement. This range suggests that while there is some consistency between the raters, there is still a noticeable amount of disagreement, which requires further investigation. After examining the dataset further, we found 12 out of 90 images to have discrepancies of 2 points or more on the 0-14 scale between the MDs. We determined, after consulting with the clinicians, that these images should be removed from our dataset, as considerable disagreement existed on their severity scores. This process resulted in a final dataset of 78 images. With all other images, $MD_1$'s scoring was used as the ground truth, as the patients interacted more frequently with $MD_1$ and were better known to them.

We also recognized that we might prefer to reduce the granularity of the classes to help interpret the severity in broader terms. Specifically, we found that the MDs themselves made no clear distinction between many pairs of successive values on the CDASI scale and that the number of images associated with a majority of the classes was quite low ($\leq 5$). Therefore, grouping the classes might help to create a "coarser" interpretation of severity and improve model performance. The clinicians had originally suggested that the scores be mapped to the four broad categories of normal, mild, moderate and severe. However, we wanted to ensure that our categories were not too narrow or broad, capturing enough variation while maintaining interpretability. Therefore, we decided to group our data in the following manner to create a balance between the number of values in each category and the significance of the severity classes. There are only two images with a CDASI score above 8, and therefore we included these two images in our most severe category to simplify our analysis.

- 0: Normal
- 1-2: Very Mild
- 3-4: Mild to Moderate
- 5-6: Moderate
- 7-8: Severe

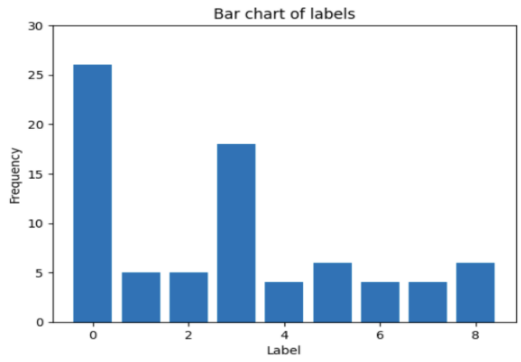


Figure 4.1: Label Distribution before grouping. Classes 0 - 8 are displayed, with Class 0 having the highest frequency, and multiple classes having less than 5 images.

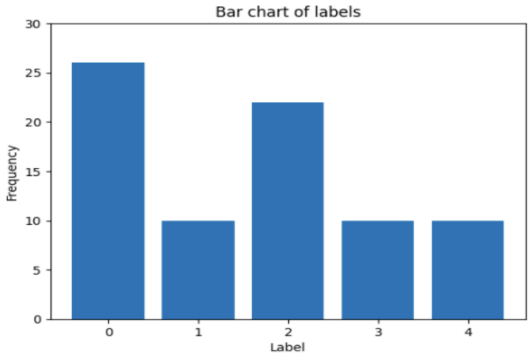


Figure 4.2: Label Distribution after grouping. Classes 0 - 5 are displayed, with a more equal distribution of labels across classes.

Figure 4.1 and Figure 4.2 display the distribution of labels before and after our final grouping. We observe that severity classes 0 and 2 have the highest frequency, while classes 1, 3, and 4 have a much lower number of labels. This class imbalance highlights the need for data augmentation, which ensures that our models receive a more balanced representation of all severity classes.

## 4.2 Dataset Augmentation

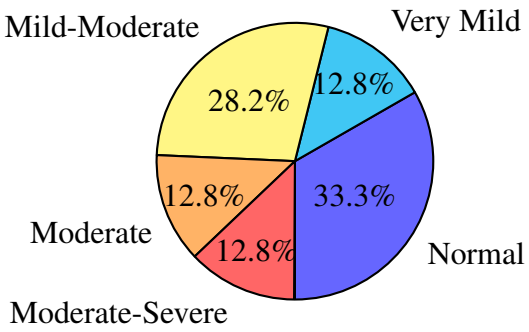


Figure 4.3: Label distribution across classes. Classes 0 and 2 have the highest frequency, while classes 1, 3, and 4 have a substantially lower number of labels.

We observe in Figure 4.3 that the percentage of the smallest minority class, the Moderate-Severe images, is approximately half of that of the Normal images. This, coupled with the small size of our dataset, prompts us to utilize data augmentation to increase the size of our dataset. Data augmentation is utilized to artificially increase the size of a dataset by applying random transformations to the original data. This can address class imbalance issues and help our models generalize better by training on more diverse examples.

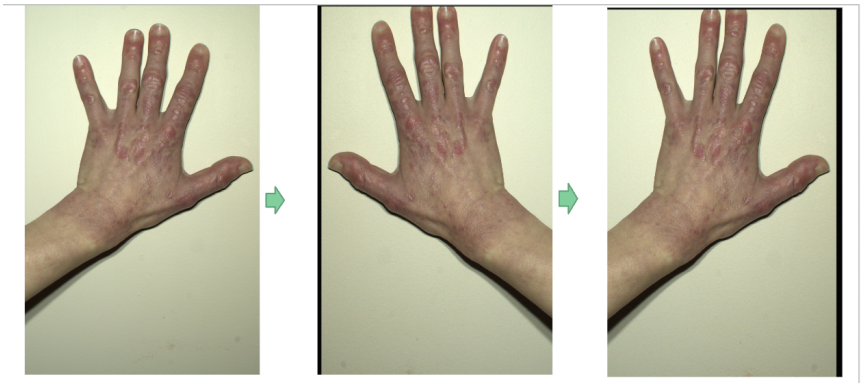

Figure 4.4: Sample augmented images. The two images on the right were produced after applying horizontal flipping and random horizontal and vertical translation to the left-most image.

| Method | Range of values |
|---|---|
| Horizontal Flip | $p = 0.75$ |
| Resized Crop | 0 - 0.5% |
| Horizontal Translation | 0 - 2.5% |
| Vertical Translation | 0 - 0.5% |
| Random Gaussian Noise | ($\mu$=0, $\sigma^2$=0.05) |

Table 4.1: Data augmentation methods. Techniques include horizontal flipping, random shifts and crops, and the addition of random Gaussian noise, all of which serve to simulate a wide range of potential real-world variations in the images.

We employed several data augmentation techniques to increase the variability of the training data and help the model generalize better to unseen examples. These techniques, seen in Table 4.1, include horizontal flipping, random shifts and crops, and the addition of random Gaussian noise, all of which serve to simulate a wide range of potential real-world variations in the images. These techniques, with parameters chosen based on practices in medical image augmentation for similar datasets, were used to increase the diversity of our dataset without significantly altering the data distribution [4]. For example, due to the consistency of lighting across the in-clinic images, transformations that altered the brightness or contrast of the images were not applied. We upsampled the minority classes until all classes were roughly equally represented in our dataset. Our dataset size increased by effectively 3 times to around 230 images post-augmentation, and this aligns with the ideal increase in dataset size according to our literature review [39].

## 4.3 Redness and Area Feature Analysis

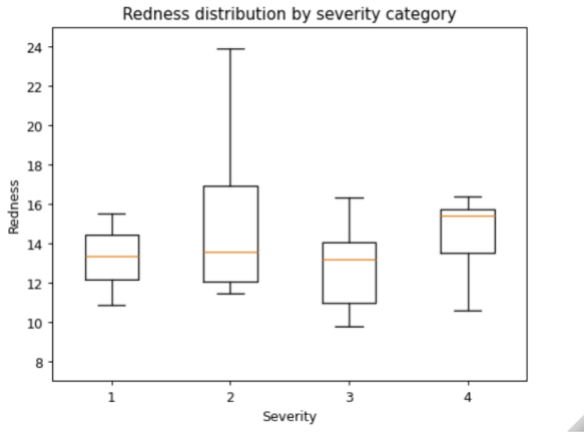


Figure 4.5: Distribution of Redness across severity classes. Displays a somewhat positive correlation between redness and severity, with a dip in average redness for class 3 and high variation in values in class 2.

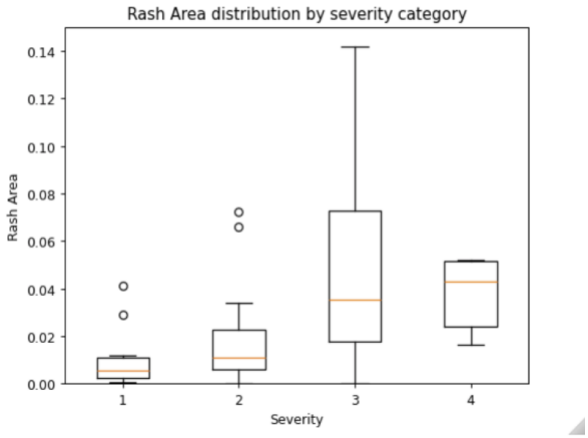


Figure 4.6: Distribution of relative Area across severity classes. Displays a more strongly positive correlation between area and severity, with high variation in values in class 3.

We conducted a statistical analysis on the rash redness values provided by the clinicians for the abnormal image classes (classes 1 through 4). The rash redness values, represented as the average $a^*$ value of the rash pixels in the $L^*a^*b^*$ color space, ranged from -128 to 128 on the red-green scale, where 0 corresponds to a neutral color and 128 indicates a strongly red hue. We plotted the redness mean and standard deviation across four severity classes, as seen in Figure 4.5.

Our analysis revealed a somewhat positive correlation between redness and severity, with greater redness values associated with higher severity classes. However, we observed a slight dip in the average redness value for class 3, which suggests some variation in how redness manifests in this particular class, perhaps due to the grouping of CDASI scores associated with this class. Additionally, it is possible that the distinguishing power of $MD_1$ for the CDASI scores of 5 and 6 was lower than other CDASI scores, leading to more inconsistencies in classifying cases within this severity class. Additionally, class 2 had a substantially greater number of images, leading to a higher deviation in the redness values for this class, likely due to the increased variability in the images.

When analyzing the relative area values (Figure 4.6), we observed a more strongly positive correlation between area and severity, indicating that as severity increases, the affected area tends to be larger. We observed a considerable spread of area values for severity class 3, similar to the redness distribution for this class. These variations highlight the inherent subjectivity and difficulty in differentiating between certain severity classes, more specifically those in the mild to moderate range. Some potential solutions to address this issue include incorporating more expert raters and aggregating their assessments, increasing the granularity of the severity scoring scale after increasing the dataset size, and predicting CDASI scores on a continuous scale.

# Chapter 5

# Handcrafted Features

## 5.1 Motivation

Our first task is to accurately capture the regions of interest in our images, which are the areas of the hands affected by CDM. We will then focus on automating the feature extraction process (previously done manually by the clinicians using the Vectra software) for subsequent classification and analysis. In image processing, thresholding is a common technique for segmenting images by classifying pixels based on their values, effectively distinguishing objects or features from the background [18]. However, determining a fixed threshold is challenging due to the high variability in the color, texture, and intensity of rashes. These characteristics can differ significantly not only between individuals but also within the same individual, influenced by factors such as skin tone, lighting conditions, and the nature of the rash. As a result, threshold values become difficult to define, limiting the effectiveness of this method.

Therefore, we sought to experiment with clustering algorithms, whereby pixels exhibiting similar redness were grouped. Among the clustering algorithms, K-means clustering is a popular unsupervised algorithm. K-means is an iterative, centroid-based clustering algorithm that partitions a dataset into similar groups based on the distance between their cluster centers [26]. If we are guaranteed that pixels representing similar colors are closer together in space than dissimilar pixels, this algorithm can be used to effectively group the pixels corresponding to the rash areas.

## 5.2 L*a*b Color Space

We first investigate which color space would be best to represent the images in. A color space is a specific organization of colors that can be mapped into a 2-D, 3-D, or 4-D coordinate system [13]. The various color spaces exist because they present color information in ways that make certain calculations more convenient or because they provide a more intuitive way to identify colors [30]. The CIELAB color space, also referred to as $L^*a^*b^*$, is a color space defined by the International Commission on Illumination in 1976. It expresses color as three values: $L^*$ for perceptual lightness and $a^*$ and $b^*$ for the four unique colors of human vision: red, green, blue and yellow. The lightness value, $L^*$, defines black at 0 and white at 100. The $a^*$ axis represents the green–red opponent colors, with negative values toward green and positive values toward red,

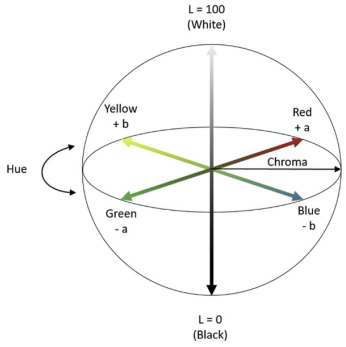


Figure 5.1: Diagram of the L*a*b space. It expresses color as three values: L* for perceptual lightness and a* and b* for the four unique colors of human vision: red, green, blue and yellow.

and the b* axis represents the blue–yellow opponent colors [12].

L*a*b* was intended as a perceptually uniform space, where a given numerical change corresponds to a similar perceived change in color [12]. This is important as in K-means clustering, the algorithm typically uses Euclidean distance to determine the similarity between data points (in this case, color values). When using L*a*b*, this distance calculation better reflects the human perception of color differences, making the segmentation more effective for tasks like distinguishing skin rashes. This color space is also useful as it decorrelates luminance, the L* channel, from chrominance information, which is what we are most interested in. This allows us to focus on the color attributes of pixels without the influence of brightness, which is crucial for accurately identifying colors associated with rashes.

The clinicians had previously identified that the pixels associated with the rash areas tend to have the highest a* values in an image. Based on this observation, we determined that applying K-means clustering to the a* and b* color channels, could effectively isolate the rash areas.

## 5.3 K-means Clustering

The K-means clustering algorithm involves an iterative process to partition data into $K$ distinct clusters. Initially, $K$ centroids are randomly selected from the dataset. For each data point, the Euclidean distance to each centroid is calculated, and the points are assigned to the cluster corresponding to the closest centroid. After this, the centroids are recalculated by computing the mean of all the data points assigned to each cluster, and this reassignment and recalculation is repeated until the centroids no longer change significantly.

Choosing the correct number of clusters, $K$, is a critical step. To determine an appropriate value for $K$, we used the Elbow method, which computes the within-cluster sum of squares across a range of $K$ values. By plotting the sum of squares against different values of $K$, we can identify the point where the rate of decrease in the sum of squares slows down, known as the ”elbow” [14]. After applying this method to a range of images, we found that a value of $K = 4$ consistently produced the best clustering results.

To avoid irrelevant pixels interfering with our K-means clustering results, we pre-process our

images by utilizing Python's rembg [10] package to isolate the hand from the background, and set the background pixels in the transformed image to be black. It is important to note that the range of values for the $a^*$ and $b^*$ channels in Python's OpenCV [9] package spans from 0 to 255. In the $L^*a^*b^*$ color space, the color black is represented with values of 128 for the $a^*$ and $b^*$ channels. Given that all patients in our dataset are Caucasian, we can reasonably assume that both the $a^*$ and $b^*$ values for normal skin and rashes will be greater than or equal to 128, and therefore that the background pixels will not be associated with our rash clusters.

From this clustering, we then selected the cluster with the highest $a^*$ centroid, as it corresponded to the region of the image most likely to represent the rash. The highest $a^*$ center value is taken to be the redness feature value and the relative area is computed by taking the ratio between the number of pixels in the highest $a^*$ cluster and the entire hand. As mentioned earlier, the three variables of relative area, redness, and relative redness, formed the initial handcrafted features we use to predict the CDASI score.

## 5.4 Classification with Handcrafted Features

We evaluated three separate classifiers to determine the most effective model for our dataset: Support Vector Machine (SVM), K-Nearest Neighbors (KNN), and Decision Tree. SVM was chosen for its robustness to overfitting, being particularly effective on small datasets [24]. KNN was selected for its simplicity and interpretability, making it easy to understand and explain the results [23]. Lastly, the Decision Tree was considered for its flexibility and ease of use with small datasets. Decision Trees can capture complex non-linear relationships and properly tuned, are resistant to overfitting [22]. To assess the performance of these classifiers, we used Leave-One-Out Cross-Validation (LOOCV) on our training set (approximately 80% of our images), which is particularly useful given the small size of our dataset. LOOCV maximizes the training data by using each data point once as a test set, providing a more accurate estimate of performance. [28]

| Number of Neighbors | | | |
|---|---|---|---|
| 3 | 5 | 7 | 9 |
| 53.9 | 49.2 | 45.3 | 41.4 |

| Max Depth | | | |
|---|---|---|---|
| 3 | 4 | 5 | 6 |
| 47.7 | 46.9 | 44.5 | 39.8 |

| C | Gamma | | | |
|---|---|---|---|---|
| | 0.1 | 0.5 | 1 | 5 |
| 0.5 | 37.5 | 46.9 | 45.3 | 50 |
| 1 | 46.9 | 46.1 | 45.3 | 46.1 |
| 5 | 49.2 | 46.9 | 49.2 | 59.4 |

Table 5.1: LOOCV accuracies for KNN (left), SVM (bottom), Decision Tree (right). We have higlighted the best hyperparameters for each classifier, determined through Grid Search.

We compute the accuracies during LOOCV and display the results in Table 5.1. To fine-tune the classifiers, we performed a grid search to determine the best hyperparameters for each classifier, ensuring optimal performance and reducing bias. After testing various combinations of parameters for each classifier, we found that applying SVM with the RBF kernel [46] and Gamma and C values of 5 produced the highest accuracy. Gamma is the kernel coefficient parameter that controls the influence of a single training example on the decision boundary and C is a regularization parameter that controls the trade-off between low training and testing error. Both parameters are vital to controlling the complexity of the classifier and its generalization capabilities.

The best-performing models yielded accuracies ranging between 45% and 60%, indicating moderate performance, with substantial room for improvement. Accuracy was chosen as the primary metric for evaluation, as it is the most relevant for clinicians in determining the model's effectiveness in real-world applications. Clinicians determined that an accuracy range of 70% to 80% was acceptable, with 80% and above considered ideal for practical use.

| | SVM | KNN | DT |
|---|---|---|---|
| Accuracy (%) | 57.3± 8.84 | 51.9 ± 7.55 | 41.5 ± 8.56 |
| Precision (macro) (%) | 55.4 ± 9.49 | 49.5 ± 8.81 | 35.4 ± 7.56 |
| Recall (macro) (%) | 55.1 ± 9.55 | 51.5 ± 9.54 | 42.9 ± 8.40 |
| F1-score (macro) (%) | 52.9 ± 8.42 | 47.3 ± 8.49 | 35.6 ± 6.48 |
| Off-by-1 acc (%) | 71.2 ± 6.49 | 68.1 ± 7.28 | 63.5 ± 8.64 |

Table 5.2: Test data metrics for SVM, KNN and Decision Tree. SVM performed the best across all metrics, with an accuracy of 57.3%. The standard deviations generally fell between 5% to 8%, indicating a reasonable level of variability that can be lowered with additional data.

We also evaluated the classifiers on held-out test sets using the best parameters identified through our grid search. We display our results in Table 5.2. To ensure robust performance, we averaged the metrics across 10 train-test (80-20) splits of our entire dataset. The standard deviations generally fell between 5% to 8%, indicating a reasonable level of variability. We expect this variation in performance metrics to decrease as additional data is added, providing more reliable and stable results.

The scores were consistently higher for SVM compared to the other classifiers. Similar to the results from LOOCV, the overall accuracy for SVM was between 40% and 60%, with a value of 57.3%. While this indicates moderate performance, an important aspect for clinicians is how close the predictions are on average to the actual outcomes. This is captured by "off-by-1" accuracy, which measures how often the classifier's prediction was one class away from the correct class. This metric is useful in clinical settings where small discrepancies in classification can still provide valuable insights. The off-by-1 accuracy for SVM was 71.2%, which is within the clinician's acceptable range of 70% to 80%, however, there is still room for improvement.

We also explored the impact of incorporating textural features into our classifier. The textural features we considered, displayed in Figure 5.2, included Contrast, Dissimilarity, Homogeneity, Correlation, and Energy, all of which were extracted using the Gray-Level Co-occurrence Matrix (GLCM) [29]. These features are particularly useful in capturing the degree of inflammation,

| | |
|---|---|
| Contrast | $\sum_{n=0}^{N_g-1} n^2 \left\{ \sum_{i=1}^{N_g} \sum_{\substack{j=1 \\ \lvert i-j \rvert = n}}^{N_g} P(i,j) \right\}$ |
| Dissimilarity | $\sum_i \sum_i \lvert i-j \rvert . P(i,j)$ |
| Homogeneity | $\sum_i \sum_j \frac{1}{1+(i-j)^2} . P(i,j)$ |
| Correlation | $\frac{\sum_i \sum_j (i-\mu_x).(j-\mu_y).(P(i,j))}{\sigma_x . \sigma_y}$ |
| Energy | $\sum_i \sum_j \{P(i,j)\}^2$ |

Figure 5.2: Textural Features, including Contrast, Dissimilarity, Homogeneity, Correlation and Energy

uniformity, and complexity of the rash. Two of the most important features, as we will see in the upcoming sections, were contrast and dissimilarity, which measure the intensity variation between neighboring pixels. By incorporating these textural features, we aimed to enhance the classifier's ability to capture subtle variations in the rash's appearance, which present differently in different classes.

| | SVM | KNN | DT |
|---|---|---|---|
| Accuracy (%) | $65.4 \pm 7.88$ | $55.4 \pm 10.6$ | $56.2 \pm 6.0$ |
| Precision (macro) (%) | $66.8 \pm 8.8$ | $57.0 \pm 9.9$ | $51.8 \pm 8.02$ |
| Recall (macro) (%) | $63.8 \pm 7.82$ | $53.5 \pm 9.67$ | $50.2 \pm 7.74$ |
| F1-score (macro) (%) | $61.6 \pm 8.50$ | $52.8 \pm 10.2$ | $47.8 \pm 6.9$ |
| Off-by-1 acc (%) | $84.2 \pm 4.69$ | $73.8 \pm 5.9$ | $79.9 \pm 5.91$ |

Table 5.3: Test data metrics with textural features for SVM, KNN and Decision Tree. All metrics improved considerably for all three classifiers. SVM performed the best across all metrics, with an accuracy of 65.4%, an 8% jump from its accuracy without textural features.

We observed that incorporating textural features led to an improvement in classifier performance, as shown in Table 5.3. For SVM, our best-performing classifier, the accuracy increased by around 8%, and other metrics showed notable improvements as well. One of the most substantial increases was the off-by-1 accuracy, which measures how often the classifier's prediction was one step away from the correct class. The addition of textural features helped reduce large errors between more distant classes, which is particularly crucial for our clinicians, as even small differences between predicted severity and the ground truth can have significant clinical implications. This improvement suggests that the spatial arrangement of pixel intensities—captured by the textural features—plays an important role in our classification task, highlighting the benefit of considering not just raw pixel values, but also the patterns and relationships between them.

# Chapter 6

# Semantic Image Segmentation

## 6.1 Motivation

We investigated the potential reasons behind the unsatisfactory results by comparing our clustering results against the actual rash areas as determined by the clinicians. For each image, we highlighted the pixels corresponding to the rash cluster from our K-means results, and set the remaining pixels to a fixed value. The clinicians then manually highlighted the rash areas on the original images using the Pixlr image editing tool (https://pixlr.com). The pixels corresponding to the rash clusters were set to 255, and the rest of the pixels were set to 0.

To assess the performance of our clustering, we computed the mean Intersection over Union (IoU) score between each K-means highlighted image and its clinician-highlighted counterpart. The IoU score measures the overlap between the predicted area and the ground truth mask, providing a quantitative assessment of how well the clustering algorithm's output aligns with the expert annotations [2]. By evaluating the IoU scores for each class, we were able to gain deeper insights into which specific classes the clustering algorithm performed poorly on and why. Our results are displayed below in Table 6.1

| | Mean IoU (%) |
|---|---|
| Class 0 | 54.6 |
| Class 1 | 61.7 |
| Class 2 | 60.9 |
| Class 3 | 64.9 |
| Class 4 | 66.8 |

Table 6.1: Mean IoU across classes. The mean IoU score across categories was generally acceptable, but the normal class had a mean IoU value of 54.6%, considerably lower than the other categories.

The mean IoU score across categories was acceptable, with all values greater than 50%, though this is not optimal. Specifically, the normal class had a mean IoU value of 54.6%, which is considerably lower than the other categories. Upon further visual inspection of the segmented images, it is apparent that K-means clustering erroneously highlights a large number of pixels for

the normal hand images. This occurs because the K-means algorithm simply selects the cluster with the highest $a^*$ value, which, in the case of a person with no rash, results in a selection of a significant number of normal skin pixels, as seen in Figure 6.1.

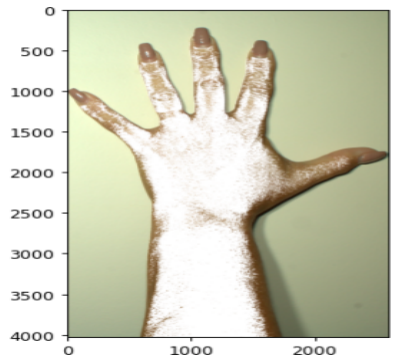


Figure 6.1: Normal hand highlighted with K-means results. For a hand image with no rash, our current clustering algorithm erroneously selects a large number of normal skin pixels.

This over-selection of pixels in the normal category contributes to the lower IoU score. Given these findings, it seems prudent to explore more complex methods to better distinguish between milder and more severe cases of CDM. This highlights the need for more precise segmentation methods to better delineate the rash areas.

To address this, we explored more advanced approaches, specifically using semantic segmentation models designed to classify each pixel in an image according to its semantic category. These models can more effectively capture the fine-grained distinctions between different regions of interest (ROI), such as the rash areas and the surrounding normal skin. Post-segmentation, the region corresponding to the rash area class will be isolated from the rest of the image and these regions will be fed into the clustering algorithms for feature extraction. We hypothesize that allowing the clustering algorithms to focus on segmented input that specifically targets the the area of the image that contains the rash, could significantly improve the downstream classifier's performance. By restricting the analysis to the ROI and extracting features only from this relevant region, we expect to reduce the influence of irrelevant background pixels, leading to more precise classifications and higher overall performance.

### 6.1.1 Ground Truth Mask Creation

In collaboration with the clinicians, we created masks for the in-clinic image dataset. First, using Python's rembg package, [10] we separated the hand from the background in the images, allowing for a more focused analysis of the relevant regions. We then leveraged the Pixlr software to manually highlight the rash area, ensuring that the highlighted areas were broad enough to account for the diverse presentations of the rash across different patients. This approach of grouping together the smaller rash areas into one large region of interest aims to improve the model's ability to generalize to various appearances of the rash.

We finally created grayscale masks with three distinct classes of pixels. An example of this is showcased in Figure 6.2. Class 0 mapped to the background, class 1 mapped to the normal

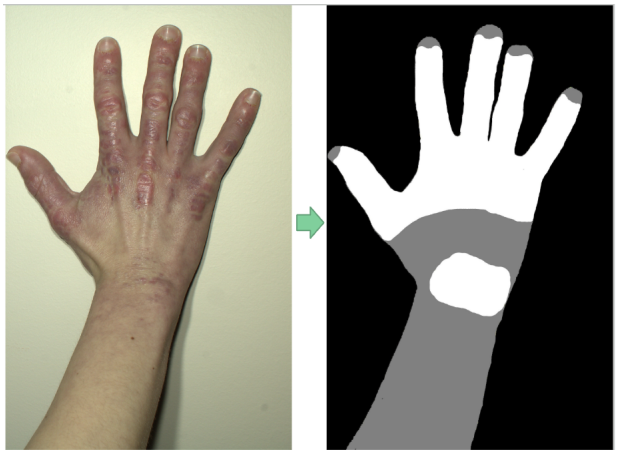

Figure 6.2: Image and corresponding ground truth mask. Masks contain three distinct classes of pixels. Class 0 mapped to the background, class 1 mapped to the normal skin pixels, and class 2 mapped to the rash area.

skin pixels, and class 2 mapped to the rash area. Through this process, we created a ”ground truth” mask, representing an accurate delineation of the rash areas identified by the clinicians. By defining these clear classes, we can better distinguish between normal skin and the rash area, ultimately improving the predictive accuracy of downstream classifiers.

## 6.2 Architecture and Training Setup

### 6.2.1 Explored Segmentation Architectures

We explored three state-of-the-art models for semantic segmentation. The U-Net model is widely used in the medical analysis domain due to its encoder-decoder structure with skip connections, which helps capture both high-level semantic features and fine-grained details, making it ideal for tasks like segmenting rash areas. [41] U-Net++, an enhanced version of U-Net, improves upon the original by utilizing nested skip pathways to learn more intricate features, improving its ability to handle diverse presentations of the rashes. [54] DeepLabV3+ utilizes an encoder-decoder structure with atrous convolutions, enabling the model to capture features at multiple scales. This is especially useful for handling complex boundaries and irregular textures, which are common in our dataset. [7] All three of these models capture varying levels of detail and complex patterns, making them highly effective for this semantic segmentation task.

### 6.2.2 Training Setup

We leveraged pre-trained backbone/encoder weights from PyTorch [38] for several advanced models, including ResNet-50, Xception, and EfficientNet-b7, all of which were trained on the ImageNet1K Dataset [11]. These pre-trained weights provided a strong foundation for our models, enabling them to leverage knowledge learned from large-scale image classification tasks, which can be transferred effectively to our medical image segmentation problem.

To evaluate model performance, we performed k-fold cross-validation with 4 folds, which allowed us to test the model's generalization capability by training it on different subsets of the data while validating on the remaining fold. For optimization, we used the AdamW optimizer with the ReduceLROnPlateau learning rate scheduler. AdamW is particularly advantageous due to its decoupled weight decay, which helps with better regularization and generalization, reducing the risk of overfitting [36]. Furthermore, the learning rate scheduler helps by reducing the learning rate when the validation loss plateaus, preventing unnecessary fluctuations in training and allowing the model to converge more efficiently [37].

The images were resized to (320, 320), and were normalized according to the ImageNet mean and standard deviation values, which helps scale the input data properly and improves convergence during training. Additionally, light data augmentation was applied during training to introduce variability in the dataset, which can help the model generalize better. Through experimentation with different initial learning rates and weight decay values, we found that the ideal initial learning rate was a low value of $1e^{-5}$. This low learning rate helped the model converge more smoothly, allowing for more stable training and better overall performance.

$$\mathbf{L}_{dice} = \frac{2 * \sum p_{true} * p_{pred}}{\sum p_{true}^2 + \sum p_{pred}^2 + \epsilon}$$

Figure 6.3: Dice Loss. This loss measures the overlap between the predicted segmentation mask and the target segmentation mask.

$$L_{\text{CE}} = -\sum_{i=1}^{n} t_i \log(p_i)$$

Figure 6.4: Cross Entropy Loss. This loss measures the difference between the predicted probability distribution and the true distribution of labels.

The loss function used was a combination of Dice Loss and Cross Entropy Loss, seen in Figure 6.3 and Figure 6.4. This summation of losses helps balance between pixel-wise accuracy and boundary alignment, encouraging more precise segmentation of the rash areas. Dice Loss, in particular, helps improve the overlap between predicted and ground truth areas, which is key to segmenting irregular or complex rash shapes. By incorporating the Dice Coefficient, the model is also encouraged to produce more balanced predictions, which is especially important in medical image analysis where our target class of pixels is often heavily underrepresented in the images [5].

We also incorporated regularization methods, including weight decay and early stopping, to further prevent overfitting and improve model generalization. Weight decay helps by penalizing large weights, encouraging the model to learn more robust features. Early stopping was used to

stop training when the validation loss stopped improving, ensuring that the model didn't overfit the training data. In the next chapter, we will explore and analyze the results of this segmentation model training process.

# Chapter 7

# Segmentation Results

## 7.1 Image Segmentation Results

| | ResNet-50 | Xception | EfficientNet-b7 |
|---|---|---|---|
| U-Net | 0.7406 | 0.7411 | 0.7242 |
| U-Net++ | 0.7345 | 0.6978 | 0.7314 |
| DeepLabv3+ | 0.7415 | 0.7359 | 0.7250 |

Table 7.1: Mean IoU across encoder-decoder combinations. The mean IoU was found to be highest for DeepLabv3+ with a ResNet-50 backbone. IoU for most encoder-decoder combinations ranged between 0.70 and 0.75.

The mean IoU score was found to be highest for DeepLabv3+ with a ResNet-50 backbone, with the mean IoU for most encoder-decoder combinations ranging between 0.70 and 0.75, as seen in Table 7.1. These scores are considered good, with acceptable IoU values typically being above 0.5, implying that our models' segmentation performance is strong. We hypothesize that DeepLabv3+ benefits from dilated convolutions, which enable the model to capture both local and global contexts, strengthening its ability to recognize complex patterns and boundaries, such as those found in irregular rash shapes. Additionally, the ResNet-50 backbone contributes to the model's high performance by providing detailed feature extraction across different spatial scales, allowing the model to effectively capture fine-grained details and large-scale features.

As discussed earlier, after segmentation, the region corresponding to the rash area class will be isolated from the rest of the image and these regions will be fed into the clustering algorithms for feature extraction. To examine the effect of segmentation on our feature distributions, we plotted the redness and relative area distributions across severity classes for images pre and post-segmentation. This can be seen in Figure 7.1 and Figure 7.2. The features were extracted from the segmented images in the same manner as described in chapter 5. We observe that the general trend in mean redness values across severity classes was somewhat positive in both cases. However, the variation in values for each severity class and the mean redness values themselves were noticeably different. To better capture the redness values for class 0, we introduced a threshold: if the segmented output area was below this threshold, we set the redness value to 128, effec-

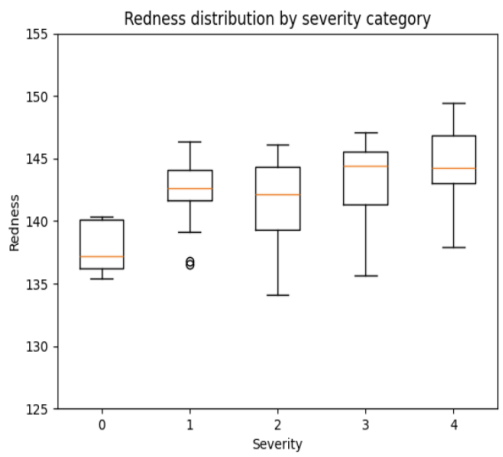


Figure 7.1: Redness Distribution before segmentation. Average redness generally increased as severity did, with large spread of values in the more extreme classes.

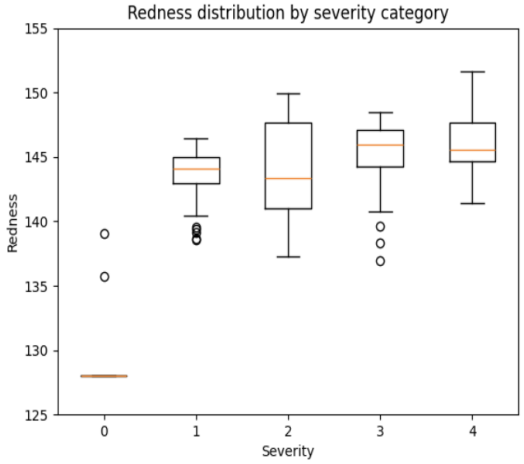


Figure 7.2: Redness Distribution after segmentation. The mean redness values for each class increased, with a large spread of values in class 2.

tively treating the image as a normal hand image and ignoring small, insignificant regions that might be present in the segmented output. This adjustment aligned well with the normal class images, as evidenced by the average redness of the normal class post-segmentation being close to 128. For the abnormal classes, the mean redness increased slightly after segmentation, as K-means focused on more relevant rash areas, where redness is typically more pronounced. This demonstrates that segmentation and K-means clustering were together able to more accurately highlight the rash regions.

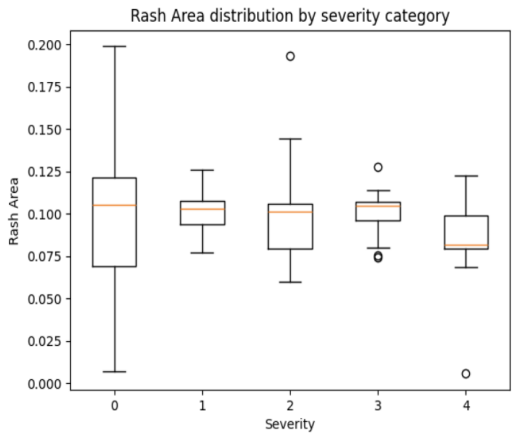


Figure 7.3: Relative Area Distribution before segmentation. No real trend observed and a large spread of values for class 0.

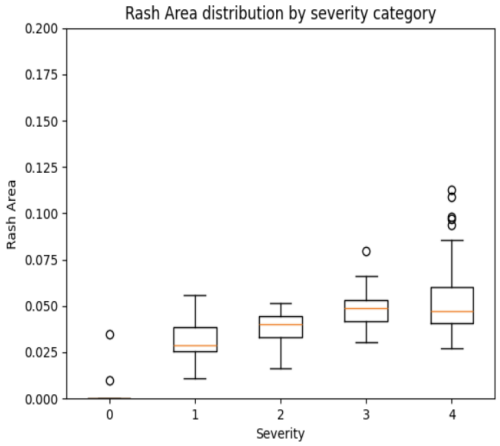


Figure 7.4: Relative Area Distribution post segmentation. Stronger positive correlation and average rash area of 0 for normal class.

We performed a similar analysis on the relative area values and observed a large variance

in the pre-segmentation values for the normal class (Figure 7.3), likely due to the presence of background noise or irrelevant regions in the image influencing the clustering output. To address this, we once again used a threshold: if the segmented output area was below a certain size, we set the relative area value to 0. This adjustment aligned well with the normal class, as evidenced by the average relative area being around 0 in Figure 7.4.

We also observed a clear, positive trend in the relative area values for the abnormal classes post-segmentation, as opposed to no clear trend existing for the pre-segmentation data. The mean relative area values decreased significantly post-segmentation as well, signifying that the model was successfully identifying and highlighting more relevant rash areas, rather than the surrounding skin as it did previously. Ultimately, segmentation ensured that regions not corresponding to the rash or abnormal skin areas did not contribute to the clustering results, which in turn is expected to refine the downstream classifier's performance.

## 7.2 Classification with Segmented Images

| | SVM | KNN | DT |
|---|---|---|---|
| Accuracy (%) | $65.4 \pm 7.88$ | $55.4 \pm 10.6$ | $56.2 \pm 6.0$ |
| Precision (macro) (%) | $66.8 \pm 8.8$ | $57.0 \pm 9.9$ | $51.8 \pm 8.02$ |
| Recall (macro) (%) | $63.8 \pm 7.82$ | $53.5 \pm 9.67$ | $50.2 \pm 7.74$ |
| F1-score (macro) (%) | $61.6 \pm 8.50$ | $52.8 \pm 10.2$ | $47.8 \pm 6.9$ |
| Off-by-1 acc (%) | $84.2 \pm 4.69$ | $73.8 \pm 5.9$ | $79.9 \pm 5.91$ |

Table 7.2: Test data metrics with textural features for SVM, KNN and Decision Tree. SVM performed the best across all metrics, with an accuracy of 65.4% and an off-by-1 accuracy of 84.2%, an 8% and 13% jump from its accuracies without textural features.

| | SVM | KNN | DT |
|---|---|---|---|
| Accuracy (%) | $76.2 \pm 5.91$ | $66.9 \pm 5.49$ | $67.3 \pm 4.94$ |
| Precision (macro) (%) | $80.1 \pm 5.35$ | $74.2 \pm 5.62$ | $71.3 \pm 7.15$ |
| Recall (macro) (%) | $76.2 \pm 5.91$ | $71.8 \pm 6.38$ | $69.7 \pm 6.93$ |
| F1-score (macro) (%) | $76.7 \pm 5.79$ | $68.8 \pm 6.52$ | $68.0 \pm 5.92$ |
| Off-by-1 acc (%) | $94.6 \pm 3.53$ | $94.2 \pm 3.94$ | $87.7 \pm 5.1$ |

Table 7.3: Test data metrics on segmented images. All metrics improved considerably for all three classifiers. SVM performed the best across all metrics, with an accuracy of 76.2%, an 11% jump from its accuracies without textural features.

Our metrics showcased segmentation producing an even greater accuracy boost than the incorporation of textural features, with this trend observed across all evaluation metrics and classifiers, as seen in Table 7.2 and Table 7.3. The SVM classifier had the highest accuracy, with its F1-score seeing the greatest improvement, highlighting how segmentation enhanced the classifier's ability to distinguish between different classes and correctly identify positive instances

while avoiding false positives. By focusing specifically on the ROI, segmentation ensures that the model concentrates on the relevant areas of the image, which is crucial for accurate classification and diagnosis. This approach significantly reduces noise from irrelevant areas, such as background or non-rash regions, allowing the model to make more precise predictions.

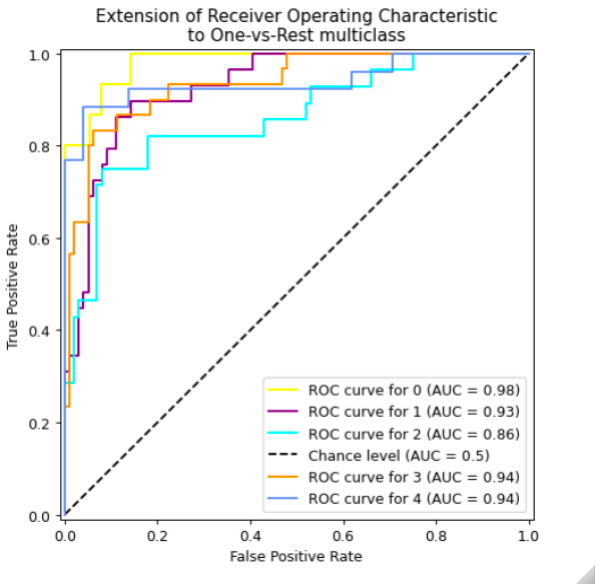


Figure 7.5: ROC curve for segmented images plotted for each class using the one-vs-rest method. Classes 0 and 4 had the highest AUC, indicating that the classifier had better discriminative ability for identifying these classes.

| Class | 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|---|
| TPR | 0.87 | 0.52 | 0.43 | 0.67 | 0.85 |

Table 7.4: TPR at FPR of 0.05 for segmented images. TPR was highest for classes 0 and 4, supporting our hypothesis that extreme classes are easier to distinguish from the rest.

We also plotted ROC curves on the held-out test data (seen in Figure 7.5), which plot the True Positive Rate (TPR) against the False Positive Rate (FPR) at various thresholds, using the one-vs-rest method. [17] This method computes the ROC curve for each class separately, each time regarding the given class as the positive class and the remaining classes as a combined negative class.

We evaluated the TPR at the FPR that the clinicians were willing to tolerate and display the results in Table 7.4. This value was determined to be 0.05. At this threshold, the TPR was highest for class 0 (normal) and class 4 (severe), supporting our hypothesis that extreme classes are much easier to distinguish when compared to intermediate classes. This was further corroborated by the higher Area Under the Curve (AUC) in Figure 7.5 for these extreme classes, indicating that the classifier had better discriminative ability for identifying these classes as expected. [17] This suggests that the classifier can more accurately classify both the absence and presence of severe

symptoms, which are typically more distinct and easier to differentiate than more subtle cases and vital to get correct for clinicians.

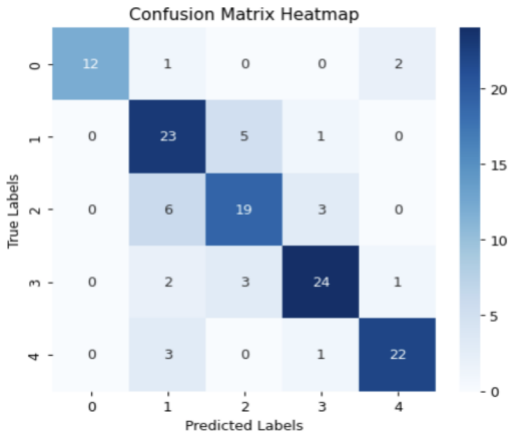


Figure 7.6: Confusion matrix on segmented image results. Classes that were most likely to be confused by the classifier were Class 1 and Class 2, as well as Class 2 and Class 3.

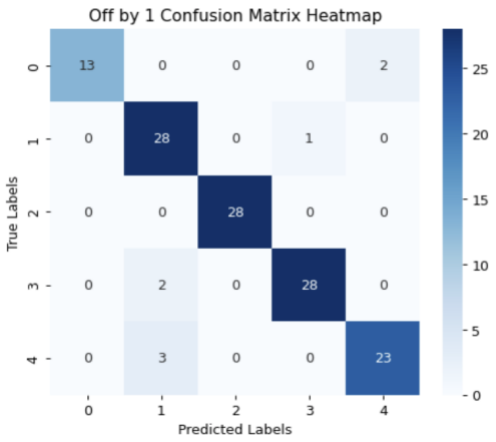


Figure 7.7: Off-by-1 Confusion matrix on segmented image results. Outliers were primarily responsible for confusing Class 0 (normal) with Class 4 (severe), as well as Class 1 (mild) with Class 4.

Through computing confusion matrices on our held-out test data, (Figure 7.6) we found that the classes that were most likely to be confused by the classifier were Class 1 and Class 2, as well as Class 2 and Class 3. This finding aligns with both our hypothesis and the analysis of the data. We recognize that the MDs did not display a clear distinction between these very similar classes when rating the images. The subtle differences between these intermediate classes may have contributed to the model's difficulty in accurately distinguishing them, as the boundaries between these categories were not sharply defined in the clinicians' assessments. These results highlight the importance of addressing class ambiguity in the data to improve classifier accuracy, particularly in cases where the severity is more nuanced.

We computed the off-by-1 confusion matrix, seen in Figure 7.7, while considering predictions that were off by one class as accurate, which allowed for a more forgiving evaluation of the classifier's performance. The analysis revealed that outliers were primarily responsible for confusing Class 0 (normal) with Class 4 (severe), as well as Class 1 (mild) with Class 4. Upon further inspection, we found that some images in Class 0 and 1 were poorly segmented, with the K-means algorithm picking up irrelevant regions in the background or non-rash areas, leading to misclassifications.

We also observed that in calculating the area and redness features, the K-means clustering algorithm highlighted a considerable number of irrelevant pixels for certain images. Images containing more distinct, well-defined, and often uneven rash regions were clustered and classified more accurately, as opposed to rashes that presented as a general reddish region. To combat this issue, we could utilize more powerful clustering algorithms or CNN-extracted features to better handle multiple types of rash presentations. Recognizing the limits of handcrafted feature

extraction and perhaps using the handcrafted features to augment the abilities of CNN-extracted features would result in more accurate and reliable classifications.

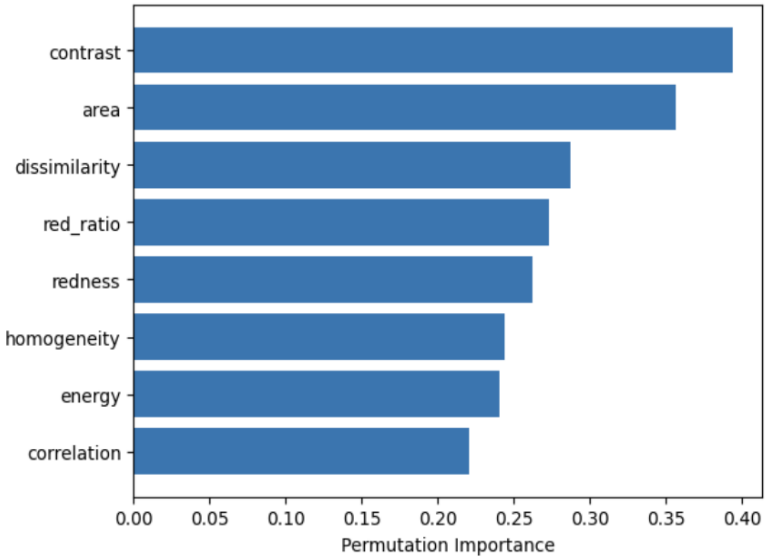


Figure 7.8: Feature Importance graph. Textural features were found to be vital to the classifier, with contrast and dissimilarity standing out as particularly impactful. Area and redness were also found to be key, reaffirming the clinicians' hypothesis that these features were important in classifying rash severity.

We were also interested in better understanding which specific features had the greatest impact on the classifier's predictive power and we visualized this by plotting a feature importance graph, [45] shown in Figure 7.8. Textural features were found to be vital to the classifier, with contrast and dissimilarity standing out as particularly impactful. Contrast and dissimilarity are valuable for identifying heterogeneous regions and textural irregularities within the images as they measure the intensity difference between neighboring pixels. This helps the classifier distinguish areas with significant variabilities, such as the rash, from more uniform regions like normal skin. Along with the textural features, area and redness also ranked quite high, reaffirming the clinicians' hypothesis that these features were highly useful in classifying rash severity.

# Chapter 8

# CNN-based Classification

## 8.1 Motivation

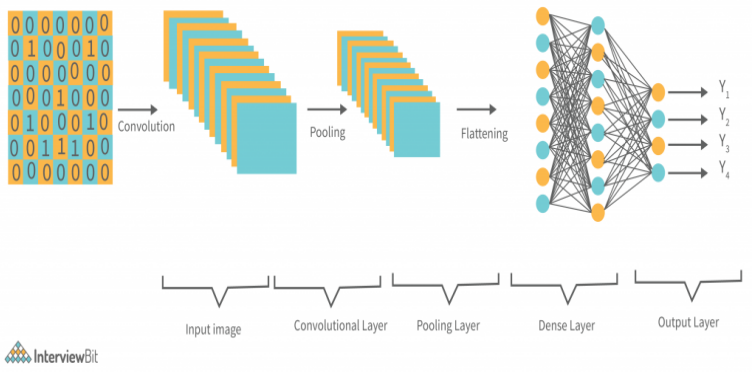


Figure 8.1: A typical CNN architecture

We recognize that while handcrafted features incorporate domain knowledge and provide a degree of explainability for clinicians, they may not be as flexible or robust in capturing the complex, abstract patterns present in medical images, especially those with subtle variations in rash appearance. Additionally, they pose a high risk of overfitting to the dataset, as they are often tailored to specific patterns in the training data and may not generalize well to unseen examples. In contrast, Convolutional Neural Networks (CNNs), particularly those with pre-trained backbones, are capable of learning multi-scale hierarchical feature representations and are capable of extracting complex and abstract features from the data. Therefore, the CNN-based features are likely to produce better classification results on our data as they capture more subtle patterns in the images and aren't overly reliant on predefined rules in the way handcrafted features might be [33] [52].

To leverage these advantages, we harnessed a pre-trained ResNet-18 model [19] for feature extraction. This model was chosen due to its high performance during our fine-tuning procedure, which will be discussed in the upcoming sections.

## 8.2 Classification with CNN-extracted Features

| | SVM |
|---|---|
| Accuracy (%) | 76.2 ± 5.91 |
| Precision (macro) (%) | 80.1 ± 5.35 |
| Recall (macro) (%) | 76.2 ± 5.91 |
| F1-score (macro) (%) | 76.7 ± 5.79 |
| Off-by-1 acc (%) | 94.6 ± 3.53 |

Table 8.1: Test data metrics on segmented images. SVM had an accuracy of 76.2%, an 11% jump from its accuracy without segmentation.

| | SVM |
|---|---|
| Accuracy (%) | 85.4 ± 3.77 |
| Precision (macro) (%) | 86.8 ± 4.62 |
| Recall (macro) (%) | 84.2 ± 2.85 |
| F1-score (macro) (%) | 83.4 ± 3.8 |
| Off-by-1 acc (%) | 94.2 ± 5.51 |

Table 8.2: Test data metrics for CNN-extracted features. All metrics improved considerably. SVM had an accuracy of 85.4%, a 9% jump from its accuracy on handcrafted features alone.

We analyze the results for only the SVM classifier, as it outperforms the other classifiers by a wide margin. The classification results for the CNN-extracted features and the handcrafted features (extracted from the segmented output) are displayed in Table 8.1 and Table 8.2. We see that the inclusion of CNN features resulted in a substantial accuracy boost, improving the SVM's performance across all metrics except for off-by-1 accuracy, which saw virtually no improvement.

We also experimented with fusing our previously used handcrafted features with the CNN-extracted features. We performed Principal Component Analysis (PCA) on the CNN features to reduce their dimensionality, and then concatenated these reduced feature vectors with the handcrafted features. However, this fusion did not result in a significant boost in accuracy or other performance metrics. This suggests that the multi-scale, hierarchical feature representations captured by the CNN likely encompassed the relevant patterns that the handcrafted features aimed to highlight, making the fusion unnecessary.

| Class | 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|---|
| TPR | 0.97 | 0.93 | 0.91 | 0.98 | 1.00 |

Table 8.3: TPR at FPR of 0.05 for CNN-extracted features. TPR was highest for classes 0, 3, and 4, supporting our hypothesis that extreme classes are easier to distinguish from the rest.

We also computed the ROC curve, seen in Figure 8.2, for each class using the one-vs-rest method [17]. Once again, for each class, we evaluated the TPR at the FPR that clinicians were

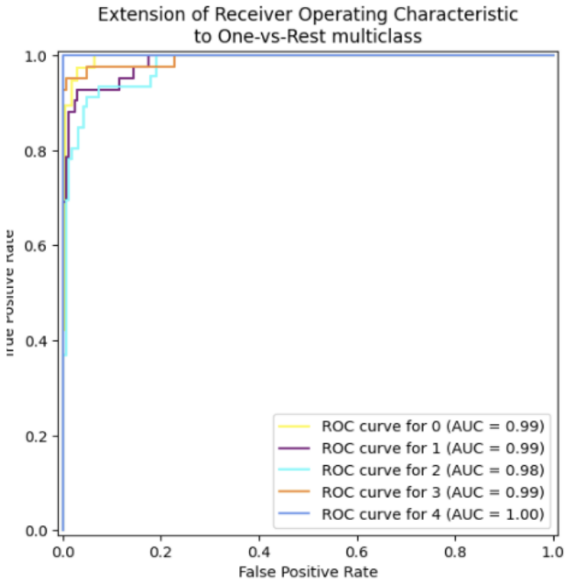


Figure 8.2: ROC curve for CNN-extracted features, plotted for each class using the one-vs-rest method. Extreme classes had the highest AUC scores, indicating that the classifier had better discriminative ability for identifying these classes.

willing to tolerate, which was determined to be 0.05. The TPR was highest for Class 0, Class 3, and Class 4, at this FPR value, as showcased in Table 8.3. This finding supported our hypothesis that extreme classes are much easier to distinguish than intermediate classes, and was further supported by the higher Area Under the Curve (AUC) values observed for these classes, seen in Figure 8.2 [17]. The model showed better performance in accurately identifying the more distinct and clearly defined classes, which is consistent with clinical observations where classes with subtler differences in severity are more difficult to classify.

Through computing confusion matrices on our held-out test data, seen in Figure 8.3, we found that the classes that are most likely to be confused by the model were Class 0 and Class 1, as well as Class 1 and Class 2. This pattern suggests that for the CNN-based features, milder cases (Class 0, 1, and 2) were particularly challenging to differentiate; as more data is added, the model's ability to distinguish between these similar classes should improve, especially with the added variety that a larger dataset can provide. However, the model performed quite well in distinguishing between Class 0, the normal class, and the extreme classes, Class 3 and Class 4, an improvement from the handcrafted features. The outliers that were present while using the handcrafted features aren't an issue for the CNN-extracted features, which is a testament to the predictive power of these features. The ability to accurately classify these extreme cases is key for clinicians, as it enables them to identify both the absence and presence of severe symptoms effectively. This performance ensures that the model is useful in clinical settings, where a clear distinction between normal and severe cases is often more critical than differentiating between milder, similar conditions.

The level of agreement between the CNN feature-based SVM model and the ground truth ($MD_1$) was measured using Weighted Cohen's Kappa, which yielded a value of $0.788 \pm 0.0736$,

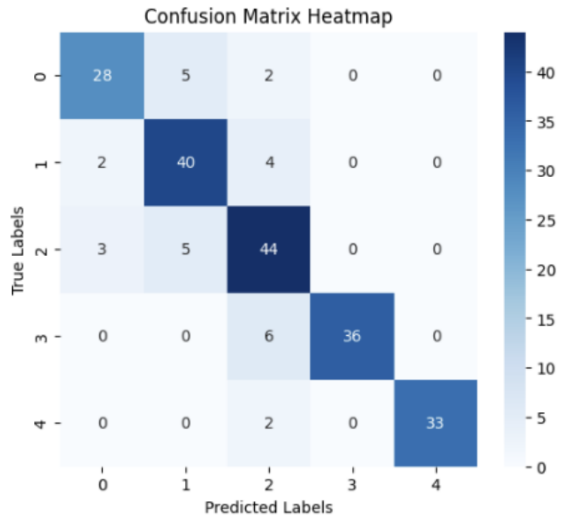


Figure 8.3: Confusion matrix for CNN-extracted features. Classes that were most likely to be confused by the classifier were Class 0 and Class 1, as well as Class 1 and Class 2, suggesting that milder classes were particularly challenging to differentiate.

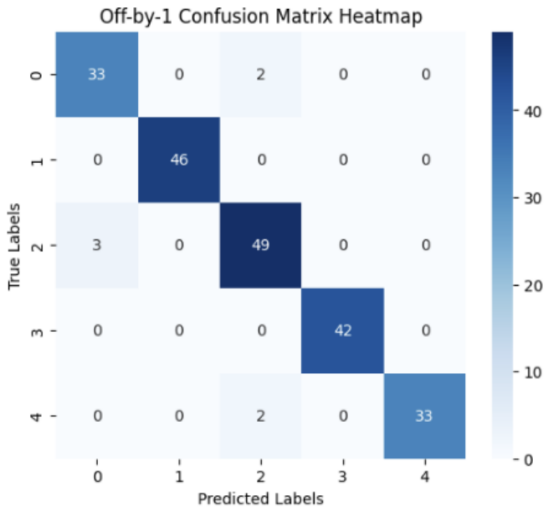


Figure 8.4: Off-by-1 confusion matrix for CNN-extracted features. Class 2 seemed to be confused with class 0 and class 4, suggesting that the discriminative power of the MDs for class 2 was particularly low.

with a 95% confidence interval of [0.644, 0.933]. This indicates substantial agreement according to Landis and Koch (1977), [27] who classify Kappa values between 0.61–0.80 as substantial. In contrast, the initial Kappa between the two MDs was 0.561 ± 0.0615, showing moderate agreement. The stronger agreement between the model and $MD_1$ suggests that the model, trained on $MD_1$'s data, aligns more closely with $MD_1$'s assessments than another human rater like $MD_2$. These results highlight the model's potential for clinical adoption, as its substantial agreement with $MD_1$ indicates it could be a useful tool in supporting clinicians.

## 8.3 Fine-tuning Setup

We evaluated the impact of fine-tuning pre-trained models on our dataset by leveraging pre-trained weights from PyTorch, specifically those trained on the ImageNet1k dataset, a diverse dataset with 1000 different categories covering a wide range of objects [11]. Fine-tuning a pre-trained model allows the model to benefit from the high number of complex features learned on a large, diverse dataset like ImageNet and adapt this knowledge to our specific task, which is crucial for capturing low-level features in the images such as edges and textures. The models that we tested were ResNet-18, ResNet-34, ResNet-50, EfficientNet-b4, and Mobilenet-v2. These models were chosen for both their high performance and for working well with limited data.

As part of our pre-processing procedure, we resized all input images to (320, 320), and normalized them based on the mean and standard deviation of ImageNet, ensuring that the input data distribution aligns with the distribution of the data the pre-trained model was originally

trained on. We applied light data augmentation techniques during training, such as random rotations and flips, to artificially increase the dataset's diversity and help the model generalize better to unseen data.

We froze the earlier layers of the pre-trained model, which capture more general, low-level features. By freezing these layers, we greatly reduced the number of learnable parameters and allowed the model to focus on learning the more task-specific layers, crucial for classifying images more accurately. For example, in ResNet-18, all layers but the last two blocks of layer 4 were frozen, leveraging the strengths of the pre-trained model while allowing it to specialize on our medical image dataset [19]. We employed k-fold cross-validation with 4 folds to compute a reliable estimate of model performance across different subsets of the dataset and mitigate overfitting.

We used the AdamW optimizer, known for handling sparse gradients and weight decay efficiently. The learning rate was dynamically adjusted during training using a scheduler, ReduceLROnPlateau, which reduces the learning rate when the validation loss plateaus, helping the model avoid overshooting the optimal minimum. The loss function chosen was Cross-Entropy Loss, which is commonly used for multi-class classification tasks and works well with the softmax output. In future iterations, a weighted loss function could be incorporated to account for the unequal distances between successive ordinal severity classes and to assign greater penalties for misclassifying certain classes than others.

To prevent overfitting, we incorporated several regularization techniques, including weight decay and early stopping. Weight decay is a form of regularization that penalizes large weights in the model, which helps prevent overfitting by encouraging simpler models. Early stopping monitors the validation loss during training and halts training when performance starts to degrade, ensuring the model doesn't overfit to the training data. We conducted a grid search to determine optimal hyperparameters for the learning rate and weight decay. Based on the search results, the best learning rate values were found to be between $3e^{-4}$ and $5^{-5}$, which provided the best balance between fast convergence and stable training. Similarly, the best weight decay values ranged between $5e^{-3}$ and $1e^{-2}$, which helped prevent overfitting while still allowing the model to adapt to the dataset.

## 8.4 Fine-tuning Results

| | Top-k Accuracy (k=1) (%) | Top-k Accuracy (k=2) (%) |
|---|---|---|
| ResNet-18 | 76.7± 7.94 | 87.9± 6.61 |
| ResNet-34 | 74.4 ± 2.87 | 89.4 ± 3.64 |
| ResNet-50 | 71.9 ± 6.06 | 85.4 ± 3.71 |
| EfficientNet-b4 | 74.2 ± 5.32 | 89.7 ± 4.58 |
| MobileNet-v2 | 75.1 ± 3.89 | 89.1 ± 6.11 |

Table 8.4: Mean Top-1 and Top-2 test accuracy for all models. We observe that ResNet-18 has the highest top-1 accuracy, with most accuracies falling between the 70% to 76% range

As seen in Table 8.4, we evaluated the average (across folds) top-k test accuracy for both k = 1 and k = 2 for each model, which is particularly useful for classes that are not easily distinguishable, providing insight into whether the model might still predict classes close to the ground truth despite not predicting the exact class. We observe that the top-1 accuracies all fall within the 70% to 77% range, with the ResNet-18 model achieving the highest top-1 accuracy. As mentioned earlier, the clinicians previously determined that an accuracy range of 70% to 80% was acceptable, which all of our model average accuracies fall in. Furthermore, the top-2 accuracies all fall between 85% and 90%, which is within the ideal range for our clinicians, indicating that our model predicts relevant classes with high probability. Further analysis could determine whether the top-2 classes are close on the severity scale, further supporting our claims that the model struggles with differentiating between subtle variations in severity.

However, when considering the k=1 accuracy, the ResNet-18 model still performed worse than our current best-performing model, which involved coupling CNN feature extraction with an SVM classifier. This discrepancy is likely due to the ResNet-18 model final layers being trained with a large number of parameters on a small dataset, which leads to overfitting and less robust performance when compared to the simpler SVM model.

## 8.5 Grad-CAM Visualization

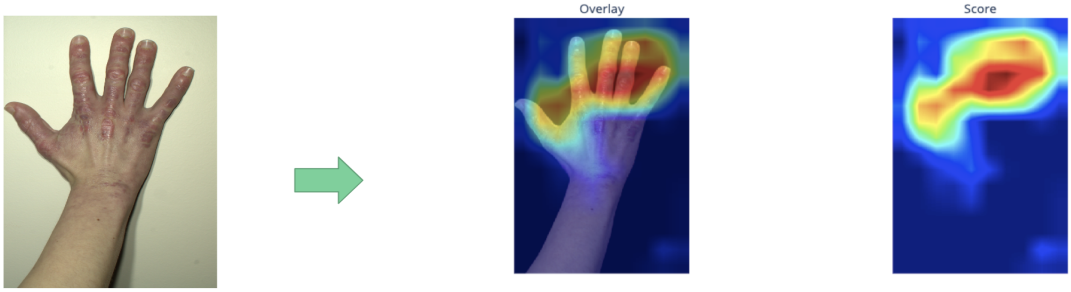


Figure 8.5: Resnet-18 Grad-CAM results. Grad-CAM highlighted the regions of the image corresponding to the fingers and knuckles of the hand, which were regions identified by the clinicians as typical areas for rashes.

Grad-CAM (Gradient-weighted Class Activation Mapping) uses the gradients of a specific class to generate a saliency map that highlights the important regions of an image for that class. [47] This technique is particularly useful for visualizing which parts of the image are most influential in the model's decision-making process. Incorporating Grad-CAM provides a degree of transparency and explainability for clinicians, which is crucial for increasing trust and ensuring that the model can be effectively integrated into clinical practice.

When applied to the fine-tuned ResNet-18 model, Grad-CAM highlighted the regions of the image corresponding to the fingers and knuckles of the hand, which were regions identified by the clinicians as typical areas for rashes. This can be seen in Figure 8.5. The fact that the model's focus aligned with the clinicians' expertise suggests that the model learnt meaningful

features that are relevant for clinical decision-making and is consistent with human judgment, which builds confidence in the model's predictions.

Finally, we hope to explore the possibility of incorporating Grad-CAM attention weights into the classification pipeline to enhance both the model's predictive performance and interpretability. We aim to first use the fine-tuned ResNet-18 model to generate Grad-CAM attention maps, and then fuse the ResNet-18 extracted features with the attention map weights. This fusion allows the model to combine both the high-level, multi-scale features learned by ResNet-18 and the spatially weighted regions emphasized by Grad-CAM. The resulting feature representation will then be passed through a final classification layer. This aims to improve the model's ability to focus on clinically relevant areas of the image and improve classification accuracy by integrating both the learned features and the regions identified through attention mapping.

# Chapter 9

# Conclusion

## 9.1 Summary

Ultimately, this work integrated a variety of methods including image preprocessing, segmentation, feature extraction, and machine learning classification techniques to create a novel, image-based approach to assessing CDM severity in patients. Automating the severity evaluation of CDM rashes will assist clinicians in tracking the progression of the disease over time for improvement after initiating treatment.

Through this work, we explored a variety of approaches for feature extraction and classification to determine the most effective method for our dataset. After a thorough evaluation, we found that CNN feature extraction combined with an SVM classifier performed the best, achieving an accuracy of approximately 85%. This approach was particularly effective at distinguishing between mild and severe classes, which is crucial for clinicians who need clear demarcation between normal and abnormal skin conditions for accurate diagnosis and treatment.

However, this study currently functions as a proof-of-concept due to the small size of our dataset. Further data collection and training are necessary to determine the robustness of our methods and enhance our model's predictive power for the more intermediate or less extreme classes, where the distinctions are subtler. Despite this limitation, the current approach demonstrated substantial agreement with $MD_1$, indicating that the model's predictions align well with clinical judgment for the more pronounced cases. This strong agreement highlights the model's potential to support clinical decision-making, and utility for remote monitoring and disease progression tracking.

## 9.2 Limitations and Future Work

One major limitation of our current dataset is the lack of diversity in skin color, which could affect the model's ability to generalize across different populations. To address this, we aim to expand our dataset to include patients with a wide range of skin tones, and address the challenges posed by darker skin tones by utilizing image-enhancing techniques to increase rash visibility and incorporating non-chromatic features such as texture. We will also evaluate our methods on similar datasets, such as Eczema images, which are more widely available in the public domain

and may offer a better range of skin tones and disease presentations than the dataset that we can procure.

As mentioned in chapter 4, another limitation of our analysis lies in the inherent subjectivity and difficulty in differentiating between certain severity classes for our expert MDs, more specifically those in the mild to moderate range. As discussed earlier, we consider potential solutions to address this issue including incorporating more expert raters and aggregating their assessments, increasing the granularity of the severity scoring scale after increasing the dataset size, and predicting CDASI scores on a continuous scale as an ordinal regression task.

The goal is to extend this work to the telemedicine image dataset, building on the reasonable results achieved so far, which support the feasibility of remote monitoring for disease detection and progression. With continued improvements, this model could become a valuable tool in supporting clinicians and enhancing patient care, especially in areas where access to specialists is limited.

Additionally, we plan to explore the use of pseudo-labelling algorithms, which can help train and generate labels on more data as it becomes available. By leveraging semi-supervised learning techniques, these algorithms can make use of unlabeled data and expert feedback to improve model performance and extend the training dataset without requiring manual labelling for every new image. As more data becomes available, we will continuously refine our models, ensuring they remain robust, accurate, and adaptable.

# Appendix A

# LOOCV Results

| K | | | |
|---|---|---|---|
| 3 | 5 | 7 | 9 |
| 0.578 | 0.555 | 0.461 | 0.453 |

| Max Depth | | | |
|---|---|---|---|
| 3 | 4 | 5 | 6 |
| 0.469 | 0.508 | 0.508 | 0.586 |

| C | Gamma | | | |
|---|---|---|---|---|
| | 0.1 | 0.5 | 1 | 5 |
| 0.5 | 0.484 | 0.593 | 0.578 | 0.461 |
| 1 | 0.578 | 0.641 | 0.664 | 0.633 |
| 5 | 0.633 | 0.679 | 0.679 | 0.648 |

Table A.1: LOOCV results with textural features for KNN (left), SVM (bottom), Decision Tree (right)

| K | | | |
|---|---|---|---|
| 3 | 5 | 7 | 9 |
| 0.688 | 0.672 | 0.633 | 0.672 |

| Max Depth | | | |
|---|---|---|---|
| 3 | 4 | 5 | 6 |
| 0.656 | 0.57 | 0.656 | 0.695 |

| C | Gamma | | | |
|---|---|---|---|---|
| | 0.1 | 0.5 | 1 | 5 |
| 0.5 | 0.484 | 0.656 | 0.679 | 0.633 |
| 1 | 0.594 | 0.688 | 0.711 | 0.719 |
| 5 | 0.648 | 0.766 | 0.773 | 0.75 |

Table A.2: LOOCV results after segmentation for KNN (left), SVM (bottom), Decision Tree (right)

# Bibliography


[1] M. Ahammed, Md. A. Mamun, and M. S. Uddin. A machine learning approach for skin disease detection and classification using image segmentation. *Healthcare Analytics*, 2: 100122, 2022. doi: 10.1016/j.health.2022.100122. URL https://doi.org/10.1016/j.health.2022.100122. 2.3

[2] Lightning AI. Mean intersection over union (miou). URL https://lightning.ai/docs/torchmetrics/stable/segmentation/mean_iou.html. 6.1

[3] C. O. Anyanwu, D. F. Fiorentino, L. Chung, C. Dzuong, Y. Wang, J. Okawa, K. Carr, K. J. Propert, and V. P. Werth. Validation of the cutaneous dermatomyositis disease area and severity index: Characterizing disease severity and assessing responsiveness to clinical change. *British Journal of Dermatology*, 173(4):969–974, 2015. doi: 10.1111/bjd.13915. URL https://doi.org/10.1111/bjd.13915. 3.2

[4] R. Attar, G. Hurault, Z. Wang, R. Mokhtari, K. Pan, B. Olabi, E. Earp, L. Steele, H. C. Williams, and R. J. Tanaka. Reliable detection of eczema areas for fully automated assessment of eczema severity from digital camera images. *JID Innovations*, 3(5):100213, 2023. doi: 10.1016/j.xjidi.2023.100213. URL https://doi.org/10.1016/j.xjidi.2023.100213. 4.2

[5] Reza Azad. Loss functions in the era of semantic segmentation: A survey and outlook, 2023. URL https://arxiv.org/html/2312.05391v1. 6.2.2

[6] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(12):2481–2495, 2017. doi: 10.1109/TPAMI.2016.2644615. URL https://doi.org/10.1109/tpami.2016.2644615. 2.1

[7] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Encoder-decoder with atrous separable convolution for semantic image segmentation. 2018. URL https://arxiv.org/abs/1802.02611. 6.2.1

[8] Mayo Clinic. Dermatomyositis, 2024. URL https://www.mayoclinic.org/diseases-conditions/dermatomyositis/symptoms-causes/syc-20353188. 1

[9] PyPI contributors. opencv-python, 2024. URL https://pypi.org/project/opencv-python/. 5.3

[10] PyPI contributors. rembg, 2024. URL https://pypi.org/project/rembg/. 5.3, 6.1.1

[11] PyTorch contributors. Imagenet dataset, 2024. URL https://pytorch.org/vision/main/generated/torchvision.datasets.ImageNet.html. 6.2.2, 8.3

[12] Wikipedia contributors. Cielab color space, 2024. URL https://en.wikipedia.org/wiki/CIELAB_color_space. 5.2

[13] Wikipedia contributors. Color space, 2024. URL https://en.wikipedia.org/wiki/Color_space. Accessed: 2024-12-11. 5.2

[14] Wikipedia contributors. Elbow method (clustering), 2024. URL https://en.wikipedia.org/wiki/Elbow_method_(clustering). 5.3

[15] DataTab. Cohen's kappa, n.d.. URL https://www.statisticssolutions.com/free-resources/directory-of-statistical-analyses/spearman-rank-correlation/. 4.1

[16] DataTab. Weighted cohen's kappa, n.d.. URL https://datatab.net/tutorial/weighted-cohens-kappa. 4.1

[17] Google Developers. Roc and auc, 2024. URL https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc. 7.2, 8.2

[18] Encord. Image thresholding in image processing, 2023. URL https://encord.com/blog/image-thresholding-image-processing/. 5.1

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. 2015. URL https://arxiv.org/abs/1512.03385. 8.1, 8.3

[20] HunterLab. What is cie lab color space?, 2024. URL https://www.hunterlab.com/blog/what-is-cielab-color-space/. 3.3

[21] IBM. Semantic segmentation. URL https://www.ibm.com/topics/semantic-segmentation. 2.1

[22] IBM. What is a decision tree?, 2023. URL https://www.ibm.com/topics/decision-trees. 5.4

[23] IBM. What is the k-nearest neighbors algorithm?, 2023. URL https://www.ibm.com/topics/knn. 5.4

[24] IBM. What is support vector machine?, 2023. URL https://www.ibm.com/topics/support-vector-machine. 5.4

[25] IBM. Image segmentation, 2023. URL https://www.ibm.com/topics/image-segmentation. 2.1

[26] IBM. K-means clustering, 2024. URL https://www.ibm.com/topics/k-means-clustering. 5.1

[27] J. R. Landis and G. G. Koch. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174, Mar 1977. doi: 10.2307/2529310. 4.1, 8.2

[28] MachineLearningMastery. Loocv for evaluating machine learning algorithms, 2020. URL https://machinelearningmastery.com/

loocv-for-evaluating-machine-learning-algorithms. 5.4

[29] MathWorks. Texture analysis using the gray-level co-occurrence matrix (glcm), 2023. URL https://www.mathworks.com/help/images/texture-analysis-using-the-gray-level-co-occurrence-matrix-glcm.html. 5.4

[30] MathWorks. Understanding color spaces and color space conversion, 2024. URL https://www.mathworks.com/help/images/understanding-color-spaces-and-color-space-conversion.html. 5.2

[31] Johns Hopkins Medicine. Dermatomyositis, 2024. URL https://www.hopkinsmedicine.org/health/conditions-and-diseases/dermatomyositis. 1

[32] Mayank Mishra. Convolutional neural networks, explained. URL https://towardsdatascience.com/convolutional-neural-networks-explained-9cc5188c4939. 2.2

[33] L. Nanni, S. Ghidoni, and S. Brahnam. Handcrafted vs. non-handcrafted features for computer vision classification. *Pattern Recognition*, 71:158–172, 2017. doi: 10.1016/j.patcog.2017.05.025. URL https://doi.org/10.1016/j.patcog.2017.05.025. 8.1

[34] National Institute of Neurological Disorders and Stroke. Dermatomyositis, 2024. URL https://www.ninds.nih.gov/health-information/disorders/dermatomyositis. 1

[35] National Institute of Neurological Disorders and Stroke. Dermatomyositis, 2024. URL https://rarediseases.info.nih.gov/diseases/6263/dermatomyositis. 1

[36] PyTorch. Adamw, . URL https://pytorch.org/docs/stable/generated/torch.optim.AdamW.html. 6.2.2

[37] PyTorch. Reducelronplateau, . URL https://pytorch.org/docs/stable/generated/torch.optim.lr_scheduler.ReduceLROnPlateau.html. 6.2.2

[38] PyTorch. Models and pre-trained weights, . URL https://pytorch.org/vision/main/models.html. 6.2.2

[39] Fazle Rahat, M Shifat Hossain, Md Rubel Ahmed, Sumit Kumar Jha, and Rickard Ewetz. Data augmentation for image classification using generative ai. *arXiv preprint arXiv:2409.00547*, 2024. doi: 10.48550/arXiv.2409.00547. URL https://doi.org/10.48550/arXiv.2409.00547. 4.2

[40] Redha Rahman, Guillem Hurault, Zihao Wang, Ricardo Mokhtari, Kevin Pan, Bayanne Olabi, Eleanor Earp, Lloyd Steele, Hywel C. Williams, and Reiko J. Tanaka. Reliable detection of eczema areas for fully automated assessment of eczema severity from digital camera images. *JID Innovations*, 3(5):100213, 2023. doi: 10.1016/j.xjidi.2023.100213. URL https://doi.org/10.1016/j.xjidi.2023.100213. 2.3

[41] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. 2015. URL https://arxiv.org/abs/1505.04597. 2.1, 6.2.1

[42] ScienceDirect. Image classification. URL https://www.sciencedirect.com/topics/engineering/image-classification. 2.2

[43] ScienceDirect. Medical image segmentation, 2023. URL https://www.sciencedirect.com/topics/engineering/medical-image-segmentation. 2.1

[44] Canfield Scientific. Vectra h1 3d imaging system, 2024. URL https://www.canfieldsci.com/imaging-systems/vectra-h1-3d-imaging-system/. 3.1

[45] scikit learn. Permutation feature importance, . URL https://scikit-learn.org/1.5/modules/permutation_importance.html. 7.2

[46] scikit learn. Rbf svm parameters, . URL https://scikit-learn.org/stable/auto_examples/svm/plot_rbf_parameters.html. 5.4

[47] Ramprasaath R. Selvaraju, Abhishek Das, C. Lawrence Zitnick, and Devi Parikh. Grad-cam: Visual explanations from deep networks via gradient-based localization. *arXiv preprint arXiv:1610.02391*, 2016. URL https://arxiv.org/abs/1610.02391. 1, 8.5

[48] Skinio. Skinio: Ai skin care, 2024. URL https://www.skinio.com/. 3.1

[49] Statistics Solutions. Spearman rank correlation. https://www.statisticssolutions.com/free-resources/directory-of-statistical-analyses/spearman-rank-correlation, 2024. 3.3

[50] J. Tiao, R. Feng, S. Bird, J. K. Choi, J. Dunham, M. George, T. C. Gonzalez-Rivera, J. L. Kaufman, N. Khan, J. J. Luo, R. Micheletti, A. S. Payne, R. Price, C. Quinn, A. I. Rubin, A. G. Sreih, P. Thomas, J. Okawa, and V. P. Werth. The reliability of the cutaneous dermatomyositis disease area and severity index (cdasi) among dermatologists, rheumatologists and neurologists. *British Journal of Dermatology*, 176(2):423–430, 2016. doi: 10.1111/bjd.15140. URL https://doi.org/10.1111/bjd.15140. 3.1, 3.2

[51] Ying Wang, Jie Su, Qiuyu Xu, and Yixin Zhong. A collaborative learning model for skin lesion segmentation and classification. *Diagnostics*, 13(5):912, 2023. doi: 10.3390/diagnostics13050912. URL https://doi.org/10.3390/diagnostics13050912. 2.3

[52] Y. Xu, L. Zhang, C. Lu, and Y. Li. A novel algorithm for multimodal biometric system fusion using svm. *IEEE Access*, 6:37142–37149, 2018. doi: 10.1109/ACCESS.2018.2846634. URL https://ieeexplore.ieee.org/document/8464688. 8.1

[53] Zhimin Zhao, Xueyan Zhang, Li Li, Wei Xu, Zhen Yan, and Fei Wang. Dermatomyositis: Pathogenesis, diagnosis, and treatments. *Journal of Clinical Neurology*, 14(1):4–13, 2018. doi: 10.3988/jcn.2018.14.1.4. URL https://pmc.ncbi.nlm.nih.gov/articles/PMC5843482/. 1

[54] Zongwei Zhou, Meng Wang, Yiming Xie, Jing Qin, and Pheng-Ann Heng. Unet++: A

nested u-net architecture for medical image segmentation. 2018. URL https://arxiv.org/abs/1807.10165. 6.2.1