Dissertation on

A Multi-stage Segmentation and Severity Classification of Diabetic Foot Ulcers using Improved Active Contour Model-based Sobel Edge Detection and Contour Analysis

Thesis submitted towards partial fulfilment of the requirements for the degree of

Master of Technology in IT (Courseware Engineering)

TILAK BHATTACHARYA
EXAMINATION ROLL NO. M4CWE24009
UNIVERSITY REGISTRATION NO. 163779 of 2022-2023

Under the guidance of

Dr SASWATI MUKHERJEE
School of Education Technology
Jadavpur University

Course affiliated to
Faculty of Engineering and Technology
Jadavpur University
Kolkata-700032
India
2024

M. Tech. IT (Courseware Engineering)
Course affiliated to
Faculty of Engineering and Technology
Jadavpur University
Kolkata, India

CERTIFICATE OF RECOMMENDATION

This is to certify that the thesis entitled **“A Multi-stage Segmentation and Severity Classification of Diabetic Foot Ulcers using Improved Active Contour Model-based Sobel Edge Detection and Contour Analysis”** is a bona fide work carried out by **TILAK BHATTACHARYA** under our supervision and guidance for partial fulfilment of the requirements for the degree of **Master of Technology in IT (Courseware Engineering)** in **School of Education Technology**, during the academic session 2023-2024.

Dr. SASWATI MUKHERJEE
SUPERVISOR
School of Education Technology
Jadavpur University,
Kolkata-700 032

DIRECTOR
School of Education Technology
Jadavpur University,
Kolkata-700 032

DEAN - FISLM
Jadavpur University,
Kolkata-700 032

M. Tech. IT (Courseware Engineering)
Course affiliated to
Faculty of Engineering and Technology
Jadavpur University
Kolkata, India

CERTIFICATE OF APPROVAL **

This foregoing thesis is hereby approved as a credible study of an engineering subject carried out and presented in a manner satisfactory to warrant its acceptance as a prerequisite to the degree for which it has been submitted. It is understood that by this approval, the undersigned does not endorse or approve any statement made, opinion expressed, or conclusion drawn therein but approves the thesis only for the purpose for which it has been submitted.

Committee of final examination ---
for evaluation of the Thesis

** Only in case the thesis is approved

DECLARATION OF ORIGINALITY AND COMPLIANCE OF ACADEMIC ETHICS

I hereby declare that this thesis contains a literature survey and original research work by the undersigned candidate as part of his **Master of Technology in IT (Courseware Engineering)** studies.

All information in this document has been obtained and presented in accordance with academic rules and ethical conduct.

I also declare that, as required by this rule and conduct, I have fully cited and referenced all materials and results that are not original to this work.

NAME : TILAK BHATTACHARYA

EXAMINATION ROLL NUMBER : M4CWE24009

REGISTRATION NUMBER : 163779 of 2022-2023

THESIS TITLE : A Multi-Stage Segmentation and Severity Classification of Diabetic Foot Ulcers using Improved Active Contour Model-based Sobel Edge Detection and Contour Analysis

SIGNATURE: DATE:

Acknowledgment

I feel fortunate while presenting this dissertation at the **School of Education Technology, Jadavpur University, Kolkata**, in the partial fulfilment of the requirement for the degree of **Master of Technology in IT (Courseware Engineering)**.

I hereby take this opportunity to show my gratitude to my mentor, **Dr Saswati Mukherjee**, who has guided and helped me with all possible suggestions, support, advice, constructive criticism, and illuminating views on different issues of this dissertation, which helped me throughout my work.

I would like to express my warm thanks to **Prof. Matangini Chattopadhyay**, **Director of the School of Education Technology,** for her timely encouragement, support, and advice. I would also like to thank **Mr. Joydeep Mukherjee** for his constant support during my entire course of work. My thanks and appreciation go to my classmates from **M. Tech in Information Technology (Courseware Engineering)** and **Master in Multimedia Development**.

I wish to thank all the departmental support staff and everyone else who has contributed to this dissertation.

Finally, I would like to express my special gratitude to my parents, who have invariably sacrificed and supported me and helped me achieve this height.

Date:
Place: Kolkata

Tilak Bhattacharya
Examination Roll Number: M4CWE24009
M.Tech. IT (Courseware Engineering)
School of Education Technology
Jadavpur University
Kolkata:70003

Contents

Executive Summary

Diabetic foot ulcer (DFU) is a common complication in patients with poorly controlled diabetes mellitus (DM). About 15% of patients with DM have a lifetime risk of developing foot ulcers. Among these individuals, 14% to 24% may require lower limb amputation due to failure to recognize and treat the ulcerated foot, which can lead to bone infection or other ulcer-related issues [1].

An in-depth review of computerized techniques for recognizing Diabetic Foot Ulcers (DFUs) has been conducted to summarize the existing research in this area. It has been determined that accurate and automatic evaluation of DFUs heavily relies on well-segmented regions, making DFU segmentation an emerging topic. Several segmentation algorithms have been developed to achieve automated detection of DFU regions.

This dissertation aims to develop an automated method for segmenting foot ulcers and classifying them according to the Wagner grading system. In the proposed approach, diabetic foot ulcer images are gathered from a Primary Care Hospital in Kolkata. A domain expert annotates the images, which are then segmented to identify the ulcer regions, and further classified for prediction. An improved Active Contour-based Sobel Edge Detection and Contour Analysis is used to segment the images and extract the region of interest (RoI). Next, MobileNetV3 is employed to classify them into the Wagner Grading system. Experiments on the collected dataset show that the proposed method achieves robust segmentation results, effectively classifying foot ulcers.

1. Introduction

1.1. Overview

Diabetes Mellitus (DM), commonly known as diabetes, is a prevalent chronic metabolic disorder characterised by prolonged elevation of blood glucose levels. Its prevalence is particularly high in developing countries [2]. According to the International Diabetes Federation, approximately 463 million adults worldwide, equating to one in 11 individuals, live with this condition [3]. Diabetic foot ulcers, a prevalent complication among diabetic patients, represent the most common lower extremity condition encountered [4]. Zhang et al.'s meta-analysis revealed a global prevalence of diabetic foot ulcers, which is 6.3% higher in men than women [5]. Foot ulcer development in diabetic patients ranges from 19% to 34% over their lifetime. The pathophysiology involves neuropathy, trauma, and often coexisting peripheral arterial disease [6]. As a result, DFU imposes a heavy clinical and economic burden on the patients and their families, particularly in underdeveloped nations. For instance, in a cohort study, high-risk DFU patients experienced toe amputation (36.4%), with mean inpatient stays ranging from 13.3 to 59.6 days and mean costs per patient-year ranging from US $3368 to US $30131 [7].

The worldwide adoption of information and communication technologies poses both challenges and opportunities for the development of contemporary healthcare systems. In recent years, computer vision algorithms have greatly extended in the area of medical imaging, including assessing different types of skin lesions, such as skin cancer and diabetic foot ulcers [8]. From a medical image analysis perspective, current ulcer image analysis addressed four tasks: high-quality image capture, wound boundary determination and localization, classification, and healing rate assessment [9]. Hence, developing automatic methods to detect, segment, and classify foot ulcers is an important localisation area of study in the computerised analysis of diabetic foot ulcers. This paper focuses on two aspects: automated segmentation for ulcer wound boundary determination and classification.

1.2. The Wagner Diabetic Foot Ulcers Grade Classification System

Foot ulcers are a common consequence of untreated diabetes that can occur due to damaged skin tissues under the big toes and on the plantar surface of the foot, which expose the underlying layer of the feet and can lead to bone damage. Effectively managing and recognising foot ulcer healing and the risk of amputation can reduce the severity of Diabetic Foot Ulcer (DFU) complications. Medical categorisation systems for diabetic foot ulcers utilize size, area, neuropathy, ischemia, and infection to predict prognosis. All of these systems rely on clinical observations and physician judgments. By considering these parameters, clinicians can better understand the extent of tissue damage, the risk of complications such as amputation, and the overall prognosis for the patient. Wagner classification, proposed by Meggitt in 1976, is one of the most widely accepted categorisation systems for diabetic foot ulcers due to its simplicity and usefulness in assessing ulcer severity. This system categorises ulcers based on factors such as depth of penetration, presence of osteomyelitis or gangrene, and amount of tissue necrosis, as shown in Table 1 [10, 11].

Table 1: Wagner DFU Grade Classification System

Ulcer Grades	**Description**
Grade-0	The skin is intact; no foot ulcer has yet developed
Grade-1	Partial superficial foot ulceration with no evidence of infection
Grade-2	No osteomyelitis or deep abscess, but a deep ulcer to bone, ligament, tendon, or joint
Grade-3	Deep foot ulcer with abscess, osteomyelitis, or joint sepsis
Grade-4	Ischemic -foot gangrene, which is present in partial gangrene of toes and forefoot
Grade-5	Entire foot gangrene

Detecting diabetic foot ulcers in foot images using computerised methods is a challenging task due to the high inter-class similarities and intra-class variations in terms of colour, size, shape, texture, and location among different types of DFUs. For a precise diagnosis of diabetic foot ulcers, a comprehensive approach involving a thorough medical history, physical examination, bacteriological investigation, blood tests, and detailed evaluation of leg blood vessels [12]. However, access to these diagnostic tools and examinations is often limited worldwide. Many computerised techniques are developed to identify and classify foot ulcers at an early stage to reduce severe complications such as amputations and mortality.

1.3. Problem Statement

A Multi-Stage Segmentation and Severity Classification of Diabetic Foot Ulcers using Improved Active Contour Model-based Edge Detection and Contour Analysis.

1.4. Objectives

The objectives are as follows:

a) To develop a novel method for segmenting the foot ulceration area in diabetic patients without using built-in segmentation tools.

b) To develop a multi-stage segmentation method that isolates the RoI using an improved Active Contour Model-based Sobel edge detection and contour analysis (iACM+SED+CA).

c) To progressively refine the segmentation process using binarization and morphological operations to isolate the RoI.

d) To optimise the hyperparameters of the Active Contour Model (ACM) to fit the ulcer boundary.

e) To classify diabetic foot ulcers in respective grades according to Wagner's Grading system using the MobileNetV3 model for its lightweight feature.

f) To perform a comparative analysis of the proposed iACM+SED+CA with other segmentation methods to assess the efficacy of the segmentation.

2. Background Concepts

Appropriate treatment of DFUs requires a comprehensive and methodical assessment. Hence, computerised evaluation of medical images is an emerging topic in computer vision and image processing. Currently, most DFU classification studies [13, 14] utilise time-consuming open-source annotation tools like CVAT, LabelImg, etc., to manually annotate the ulcer region, which is heavily dependent on the observer's perception and often leads to inaccurate predictions [15]. DFU segmentation refers to detecting and localising the boundary between the ulcer region and the healthy skin, which is a key step for further analysis and classification of foot ulcers. However, many existing studies on automatic DFU prediction still rely on built-in segmentation tools [13, 14, 16]. Accurate analysis of DFU highly relies on well-segmented ulcer regions. As a result, automatic DFU segmentation that can accurately identify and isolate the ulcer region from the background draws a lot of attention from researchers. The Active Contour model has been widely used in image segmentation for the past three decades due to its robustness, high-level knowledge integration, and strong self-adaptability. However, it relies highly on the initialisation point and suffers from limited range and noise sensitivity. Researchers have proposed several improvements over the ACM model to overcome its shortcomings [17, 18].

The Sobel operator is a classic edge detection algorithm that increases segmentation quality [19]. MobileNetV3-Small is a lightweight CNN-based model tuned for mobile phone CPUs, and it has achieved robust classification performance [20].

2.1. Canny Edge Detection

The Canny Edge Detector [21] is a multi-stage edge detection algorithm proposed by J. F. Canny in 1986. Its three general criteria are as follows.

- Detecting the edge with a low error rate
- Detected edge points should accurately localise on the centre of the real edge.
- Edges should only be marked once in the image (Minimal response).

There are 4 operational blocks in the Canny Edge Detection algorithm.

1. Smoothing using a Gaussian filter for noise removal.
2. Image intensity gradients calculation.

3. Apply non-maximum suppression to remove spurious response to edge detection.
4. Apply thresholding to determine the edges. If the pixel gradient value exceeds the upper threshold, it is accepted as a strong edge. However, if the gradient value is lower than the threshold, the corresponding pixel will be rejected. If the gradient value is between lower and upper thresholds, it will be accepted as an edge only if connected to a pixel above the upper threshold.

2.2. Active Contour Model

An active contour model, or the Snake model, uses internal and external energy forces and constraints to segregate the region of interest (ROI) from the image's background for further processing and analysis. Contours can be defined as a continuous collection of points that form the boundary of an area of interest in an image.

Kass et al. [22] proposed the Snake model, which relies heavily on the initial selection of edge points and does not achieve automatic segmentation of complex objects. The complexity of the segmentation process is directly proportional to the number of sampling points in the initial contour. External energy represents forces derived from the image itself, such as gradient and region-based forces, that control the positioning of the contour on the image. Internal energy, on the other hand, governs deformable changes within the contour, such as curvature and smoothness, which prevent it from deviating excessively from its initial position.

The energy function of the Snake model includes both external and internal energy.

$$E_{snake} = \int_0^1 \left(E_{internal} + E_{image} + E_{external}\right) ds \tag{1}$$

$$E_{external}^{(1)}(x, y) = -|\nabla I(x, y)|^2$$

or

$$E_{external}^{(2)}(x, y) = -|\nabla [G_\sigma(x, y) * I(x, y)]|^2 \tag{2}$$

$$E_{internal} = \alpha(s)|C'(s)|^2 + \beta(s)|C''(s)|^2 \tag{3}$$

In Eq. (1)

- $E_{internal}$ is the internal contour energy.
- E_{image} is the image energy.
- $E_{external}$ is the external energy.

In Eq. (2)

- $I(x, y)$ is the given continuous function of the (x,y) coordinates of the grayscale image.
- ∇ is the gradient operator.
- is the convolution operator.
- $G_{\sigma}$is a Gaussian filter with standard deviation σ.

In Eq. (3)

- *C(s)* = [x(s), y(s)], s ∈ [0, 1][0, 1] represents the evolution curve contour parameterisation of the selected region of the target image.
- $\alpha(s)$ is the first-order continuity weighting function and $\beta(s)$ is the second-order continuity weighting function of *C(s)*.
- $C'(s)$ and $C''(s)$ are the first and second derivatives of *C(s)* with respect to s.

Combining these three energies guides the active contour to converge toward the desired object boundary. This optimisation process involves iterative adjustments of the contour to balance the internal smoothness constraints and the attraction to image features.

2.3. Sobel Operator

The Sobel operator, an edge detection method proposed by Sobel and Feldman [23], is a gradient-based method which searches for strong changes in the first derivative of an image. This technique detects edges at the points with maximum gradient value and follows a criterion for determining edge pixels of the image. The Sobel operator uses two 3x3-sized kernels to estimate the gradients in the x and y directions of the image. The two kernels and their transpose are convolved with the image to horizontally and vertically approximate the derivatives to calculate gradient magnitude and direction to derive the edge. The two kernels used in the Sobel operator are shown in Eq. (4) and Eq. (5).

$$K_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} \tag{4}$$

$$K_y = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \tag{5}$$

Let I be the original image, the gradients in the x and y directions are defined in Eq. (6) and Eq. (7) which will be used in Eq. (8) to calculate the gradient magnitude G.

$$G_x = K_x * I = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} * I \tag{6}$$

$$G_y = K_y * I = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} * I \tag{7}$$

$$G = \sqrt{G_x^2 + G_y^2} \tag{8}$$

2.4. Evaluation Metrics

Evaluation Metrics are used to evaluate the performance of machine learning and deep learning models. Accuracy is the most commonly used evaluation metric representing the number of correctly predicted samples over the total number of samples in the dataset. Precision represents the proportion of the predicted samples relevant to a class. Recall or sensitivity represents the model's ability to find all relevant samples of a class in the dataset. F1-Score in Eq. (12) is the harmonic mean of Precision and Recall.

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \tag{9}$$

$$\text{Precision} = \frac{TP}{TP+FP} \tag{10}$$

$$\text{Recall} = \frac{TP}{TP+FN} \tag{11}$$

$$\text{F1-Score} = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{12}$$

In Eq. (9), Eq. (10) and Eq. (11), TP denotes the number of true positives, TN denotes the number of true negatives, FP denotes the number of false positives and FN denotes the number of false negatives.

3. Literature Survey

Heras-Tang et al. [24] proposed a novel pixel features-based two-step hybrid algorithm using logistic regression, DBSCAN clustering and morphological operators for DFU region segmentation. First, four classifiers are trained to classify pixels belonging to the ulcer region, and then the misclassified pixels are isolated using DBSCAN clustering. Morphological operations are used to refine the segmented ulcer region further. They obtained a Jaccard Index of 0.81, an accuracy of 0.94, a sensitivity of 0.86 and a precision of 0.91.

Wang et al. [9] proposed a two-stage model to determine wound boundaries. The authors used an image capture box to capture image data and employed a SLIC algorithm to segment the images into super-pixels using the SLIC algorithm. A two-stage SVM classifier is applied to determine wound boundaries. Although their approach obtained a sensitivity of 73.3% and a specificity of 94.6%, it has not been applied to a more substantial dataset.

A deep learning framework based on MobileNetsV2 [25] is proposed with a recall of 89.97%, precision of 91.01% and a mean Dice score of 90.47%.

The PSO optimization-based technique [26] is based on PSO colour-based segmentation that detects wound boundaries and extracts textual features using GLCM. Subsequently, Naive Bayes was employed to classify the wounds based on their textual features. The technique achieved an accuracy of 90.90%.

Niri et al. [27] proposed a super pixel-based CNN architecture which utilised U-Net with various morphological operations for segmentation and extracted super-pixels using SLIC. Subsequently, a state-of-the-art DNN model is employed for classification, achieving an accuracy of 92.68% and a Dice score of 75.74%.

An Artificial Neural Network (ANN) was employed for classification [28], resulting in an error rate of less than 8%. A capture protocol was used to ensure image quality. Subsequently, the images were transformed into the L*a*b colour space, and k-means clustering was utilised to segment the tissues based on colours.

In a recent study, a specialized deep learning-based classifier for diabetic foot ulcers, called DUTC Net, was introduced. The researchers preprocessed the dataset using reflection and hair removal techniques. They then used an active contour model to segment the ulcer area in foot images and trained and tested the resulting wound images using the DUTC network. The study involved two phases: in the first phase, tissue maps were generated, and in the second phase, these maps were used for prediction. The results showed a sensitivity of 97.9%, a specificity of 98%, and an accuracy of 97.2% for average tissue classification. Additionally, the study found a sensitivity of 97.1%, a specificity of 98%, and an accuracy of 97.5% for average stage classification.

In [30], the image structure adaptive gradient vector flow (ISAGVF) was used to enhance the active contour model by adjusting the smoothness constraint of the GVF Snake model. The introduction of the image structure tensor allowed the model to exhibit anisotropic behaviour. The experiment demonstrated that the model's segmentation results surpassed that of other state-of-the-art models in terms of preserving weak edges and achieving deep concavity convergence, while also possessing an expanded capture range and being less sensitive to initialization.

An active Contour Model based on Jeffreys divergence (KJD) and K-medoids clustering has been proposed [20]. In this model, the traditional Euclidean distance is replaced with Jeffreys divergence theory. The KJD energy function is created by combining this divergence with two clustering vectors generated by K-medoids clustering. To normalize the ranges of the optimized data-driven terms and the level set function, a regulation function is applied. In addition, a pre-trained YOLOv5 is used to automatically label the initial contour. When compared with other popular models, the results demonstrate that the KJD model can accurately adapt to initial contours at different positions while fitting expertly at the image boundary. It's important to note that this model is only suitable for double-phase images.

In [31], a hybrid methodology was developed for breast segmentation using adaptive histogram equalization, inversed morphology snake, and extended Stein’s unbiased risk estimator (eSURE). The authors trained a deep denoiser using eSURE to eliminate noise from the image and utilized adaptive histogram equalization to enhance image contrast. They also designed an inversed Gaussian gradient for morphology snakes to isolate the chest region from the MRIs. The framework was tested on breast MRI 3.0 T and breast MRI 1.5 T and compared with other popular approaches. The results showed a recall of 0.94, a specificity of 0.98, a dice score of 0.85, and an accuracy of 0.98.

In another study [32], a 2D OTSU multi-threshold image segmentation algorithm for inland ships was proposed. The algorithm's performance was enhanced by improving the crossover and mutation rates and employing an improved genetic algorithm to reduce the time complexity of the 2D OTSU model. The experimental results showed that this approach achieved a peak signal-to-noise ratio (PSNR) value 13.3768 higher than the average of the standard 2D OTSU algorithm while also reducing the image segmentation time.

In a recent study [33], researchers introduced a novel automated algorithm called GA11. This algorithm uses a genetic algorithm to select the best segmentation technique, colour space, and optimal parameters for image datasets. The genetic algorithm is designed to evaluate and select the best individual and then perform crossover and mutation operations to improve feature extraction. The algorithm was tested on four popular datasets, and the results showed that GA11 outperformed classical techniques, producing high-quality segmentation masks even for images with poor lighting conditions.

In [34], EU-Net, a U-Net neural architecture search utilizing a differential evolutionary (DE) algorithm, is proposed for medical image segmentation. They introduced a fully automated neural architecture search method, a variable-length strategy for optimal neural architecture depth, and a novel encoding strategy to improve the search process. The DE algorithm was used to optimize the U-Net-based architecture. Experimental results demonstrated that EU-Net improved the mIou metric by at least 6% compared to the original U-Net. Comparisons with other state-of-the-art neural architectures on the CHAOS and BUSI datasets showed that EU-Net outperformed the selected competitors regarding segmentation accuracy.

4. Proposed Approach

The proposed work aims to develop a new segmentation method that will create a bounding box and gradually isolate the foot ulcer region (RoI) of a diabetic patient. This will be followed by a classification model that categorises the ulcers into different grades according to the Wagner Grading System. The diabetic foot ulcer images are obtained from a Primary Care Hospital in Kolkata, and the ulceration regions of the images are annotated by a clinical expert. The multi-stage segmentation method involves morphological operations, improved active contour model-based Sobel edge detection, and contour analysis (iACM+SED+CA) to create a bounding box around the RoI and extract the ulcer regions accurately. A deep learning prediction model is used to classify foot ulcers into severity grades ranging from 0 to 5 from the segmented RoIs. These grades correspond to various levels of ulcer progression, from pre-ulcerative lesions to whole-foot gangrene. The bounding box descriptions are then saved in XML files according to the VOC format for the subsequent classification stage. For feature extraction and classification, MobileNetV3-Small is used to localize and classify the ulcer from the detected RoI of the DFU image from the saved XML files.

The image dataset is enhanced by cropping, horizontal and vertical flipping, and rotation to prevent overfitting. Morphological operations are used to refine the binary mask for segmentation. An improved Active Contour Model (ACM) is utilised to locate and identify the ulcer regions. The Sobel operator is applied within the bounding box defined by the improved ACM to enhance the edges and improve the quality of detected features. This is followed by contour analysis, which accurately segments the precise ulcer region and provides the Region of Interest (RoI) for classification. For feature extraction and classification, MobileNetV3-Small is used to accurately localize and classify the ulcer into six different Wagner grades from the detected RoI of the DFU image. The novelty of the proposed approach lies in the automatic segmentation method to isolate the RoI for effective classification without using any built-in segmentation tools.

The framework of the proposed (iACM+SED+CA) and the MobileNetV3-Small classifier is presented in Figure 1.

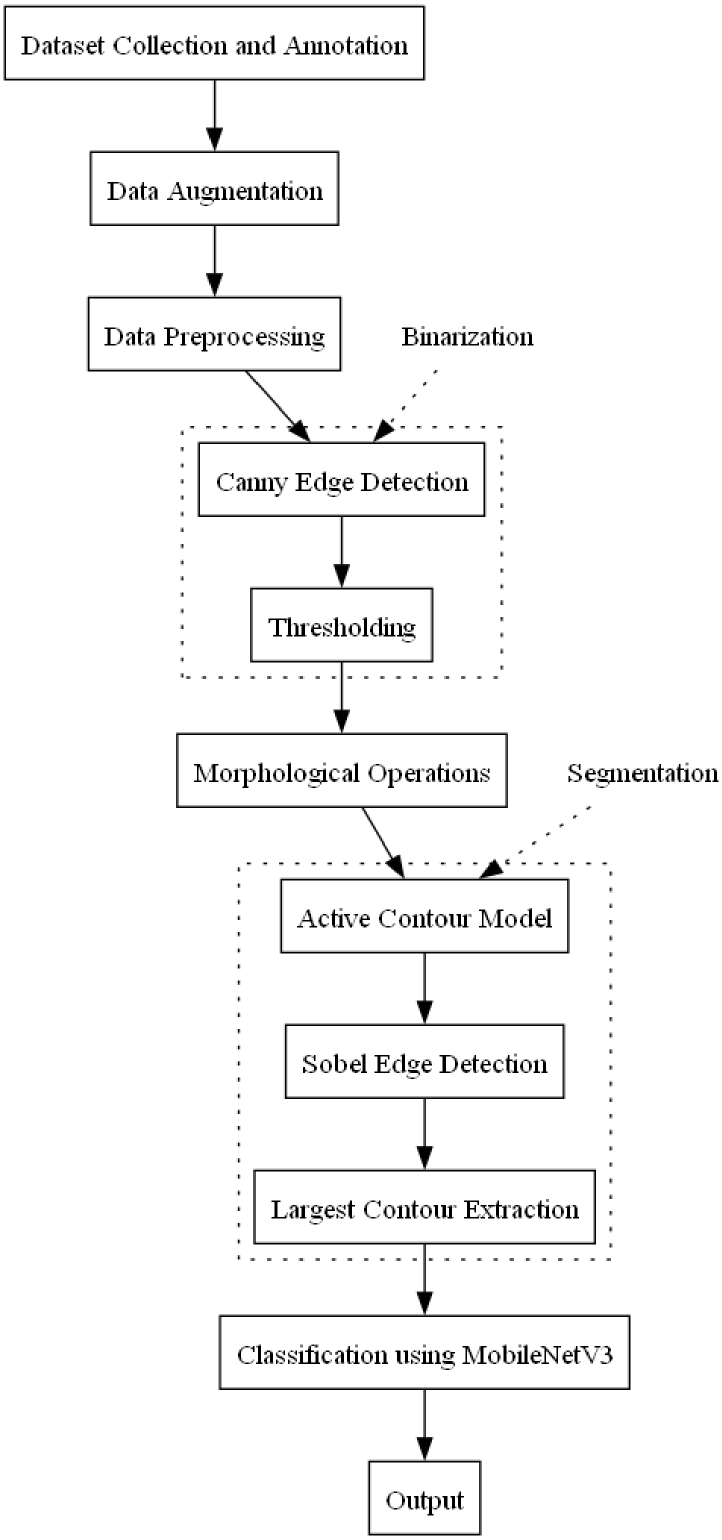


Figure 1: Framework of the proposed (iACM+SED+CA) method and MobileNetV3 classifier

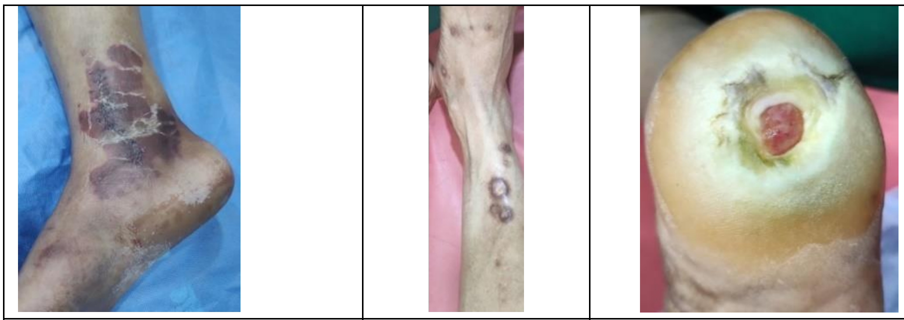
4.1. Dataset Description

The diabetic foot ulcer images are gathered from the Primary Care Hospital in Kolkata. To protect patient confidentiality, the identities of all patients are anonymised using unique identification numbers in the dataset. 923 foot ulcer images of the diabetic patients were collected, covering Grade 0, Grade 1, Grade 2, Grade 3, Grade 4, and Grade 5, each corresponding to their respective classes in Wagner's grading system, as depicted in Figure 2. The number of ulcer images for each class according to the Wagner grading scale is presented in Table 2.

Table 2: Number of DFU images in different grades

Ulcer Grades	Description	Number of Images
Grade-0	Pre foot ulceration	15
Grade-1	Partial foot ulceration	177
Grade-2	Deep ulcer to bone, ligament, tendon, or joint	491
Grade-3	Deep foot ulcer with osteomyelitis	119
Grade-4	Ischemic -foot gangrene	97
Grade-5	Entire foot gangrene	30

All the collected foot ulcer images have diverse pixel dimensions, ranging from 576x1040 to 4608x4608. The obtained dataset is graded by a clinical expert using the Wagner grade classification system, as shown in Figure 2.



Grade 0 **Grade 1** **Grade 2**

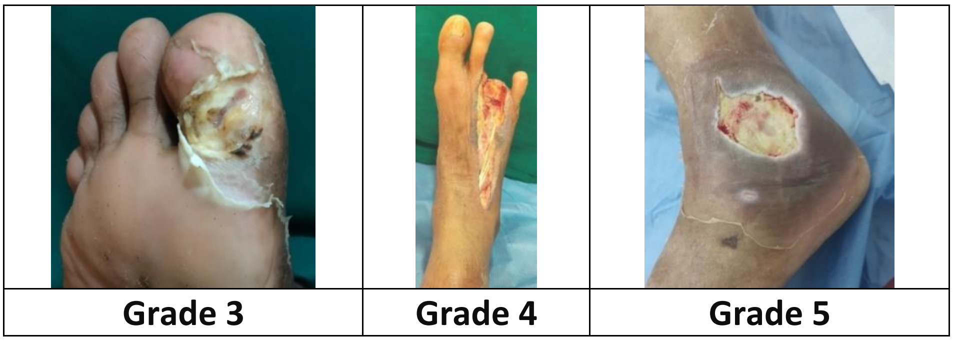

Grade 3 Grade 4 Grade 5

Figure 2: Annotation of ulcer images as per Wagner grades

4.2. Data Preprocessing

4.2.1. Data Augmentation

Due to the limited dataset size of 923 image samples, data augmentation techniques such as cropping, random rotation, and horizontal and vertical flipping produce 4987 samples. These techniques help enhance the method's generalisation ability by creating a diverse set of training samples.

4.2.2. Rescale Image Dimension

All images are resized to a standard dimension of 300x300 pixels to reduce computational load and enhance efficiency. This step also ensures uniform input dimensions for the subsequent stages.

4.2.3. Grayscale Conversion

After resizing, each image is converted to grayscale to simplify the pipeline. Different tissues have varying grayscale values, and converting the image to grayscale reduces computational complexity and allows subsequent steps to focus on the variations of intensity values, which are crucial for detecting ulcer regions.

4.3. Binarization

4.3.1. Canny Edge Detection

The grayscale image is processed using a canny edge detector to identify the potential boundaries of ulcers. This edge detection method uses intensity gradients to create smooth and strong edges. The detected boundaries generate edge maps, which emphasise the main features in the image. These edge maps are then used in the following steps, as illustrated in Figure 3.

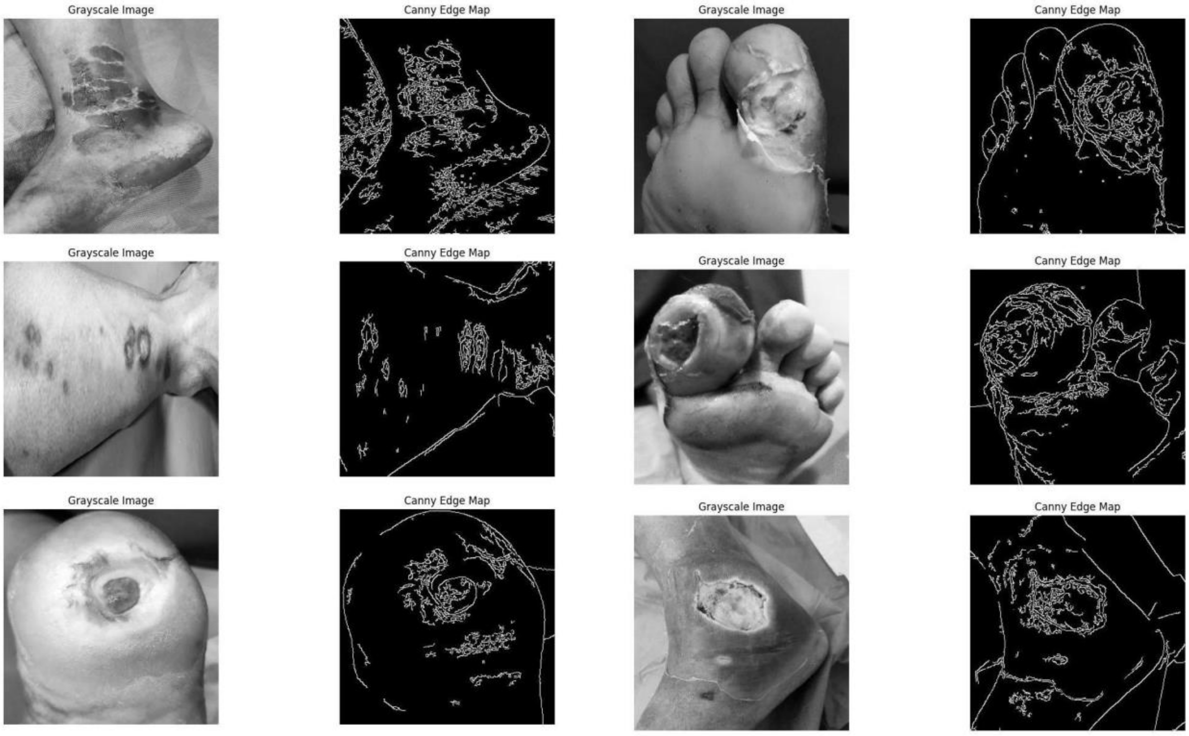


Figure 3: Edge Maps of the ulcer images after applying Canny edge detection

4.3.2. Thresholding

The Canny edge detection produces an edge map, which is then converted into a binary mask using thresholding. This helps simplify the segmentation process by focusing on regions with high gradient intensities. We use a threshold of 80: pixels with values greater than 80 are set to 255 (white), while pixels with values less than or equal to 80 are set to 0 (black). This results in a binary image where the edges detected by the Canny algorithm are highlighted as white lines on a black background. This simplifies further processing and improves efficiency.

4.4. Morphological Operations

Morphology involves using techniques to remove noise from the binary image and improve its quality, making it easier to process in later stages. This is achieved through erosion and dilation operations. Erosion works by shrinking the boundaries of foreground objects, smoothing out small irregularities, and reducing the features of the image. On the other hand, dilation is used to enhance the features of the image. To accomplish this, a closing operation followed by an opening operation is used [35, 36].

Let I_b be the binary image and S be the structural element. Erosion (E) and Dilation (D) are defined in Eq. (13) and Eq. (14).

$$E\ (I_b, S)\ =\ I_b \ominus S \tag{13}$$

$$D\ (I_b, S)\ =\ I_b \oplus S \tag{14}$$

Closing Operation

The closing operation involves applying a dilation followed by erosion. It is primarily used to fill in small holes and gaps within the foreground objects of the image, which helps create a more cohesive representation of the target features and enhances the overall quality of the binary image. The closing operation can be expressed as $I_b \bullet \mathrm{S}$ as shown in Eq. (15).

$$I_b \bullet \mathrm{S}\ =\ (I_b \oplus S) \ominus S \tag{15}$$

Opening Operation

The opening operation starts with erosion, followed by dilation. This process helps to reduce noise without affecting the overall structure of the target objects. Erosion removes small white noise components, but it also shrinks the objects. Dilation is then used to expand the object area of the eroded image without reintroducing the removed noise elements. This ensures the removal of thin lines from an image while preserving the shape and size of larger potential ulcer regions in the binary image. The opening operation can be expressed as $I_b \circ S$ as shown in Eq. (16).

$$I_b \circ S =\ (I_b \ominus S) \oplus S \tag{16}$$

In addition, the use of opening and closing operations effectively eliminates noise and enhances the key structures of the image, refining the binary images for further processing, as shown in Figure 4.

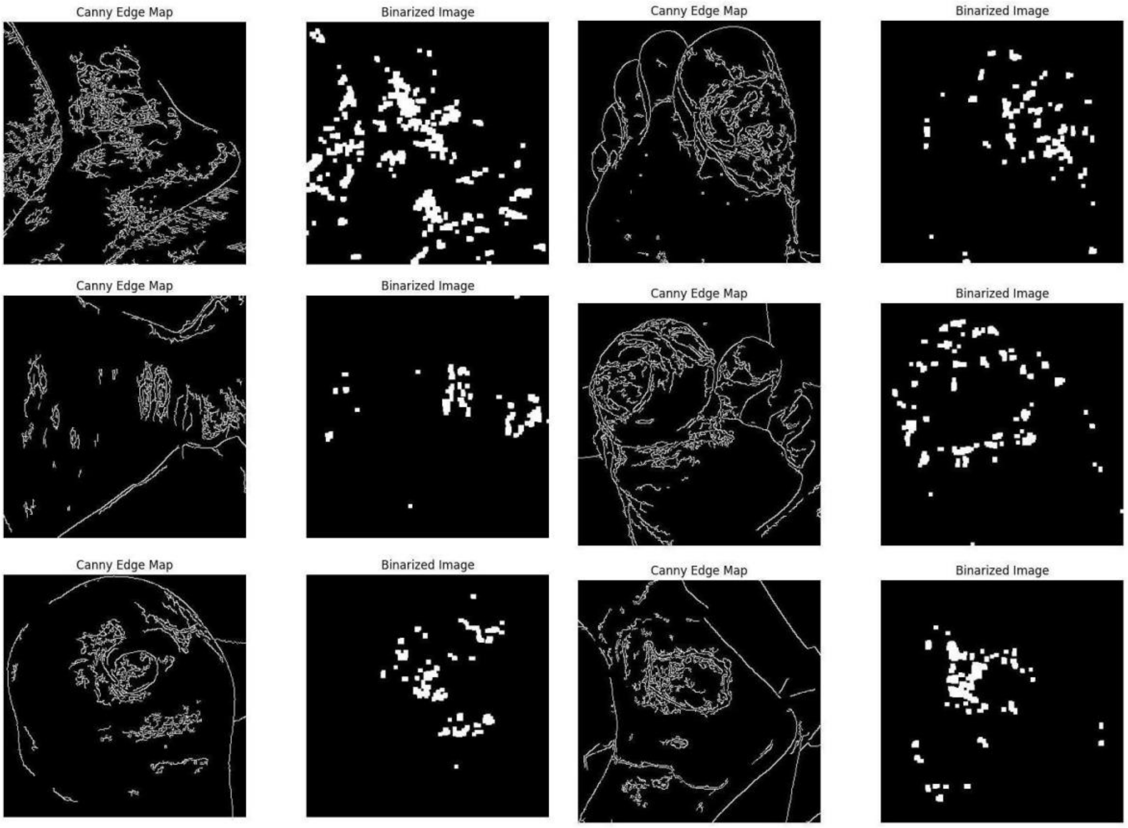


Figure 4: Edge Map after Binarization and Morphological Operations

4.5. Segmentation

4.5.1. Improved Active Contour Model (iACM)

The improved Active Contour Model (iACM) is used to iteratively adjust the contour by minimizing an energy function that balances the internal energy (contour smoothness) and the external energy (driven by the image gradient). This process allows the contour to accurately fit the boundaries of the target object and isolate the potential ulcer region. To initialize the contour, image moments are used around the centroid of the binary mask in a circular shape [37]. This enables the contour to be dynamically initialized at the centre of the mask of objects represented by the white pixels. Let's denote the spatial moments of the noise-removed binary mask obtained from the morphological operations of the previous step as M. The centroids (C_x,C_y) are obtained by calculating the spatial moments of the image.

$$C_x = \frac{M_{01}}{M_{00}} \tag{17}$$

$$C_y = \frac{M_{10}}{M_{00}} \tag{18}$$

M_{00} is the zero-order moment, and M_{01}, M_{10} are first-order moments in Eq. (17) and Eq. (18). The contour is initialised around the obtained centroid in a circular shape with the radius R. The coordination of the points on the initial contour $(r(\theta), c(\theta))$ can be calculated as follows.

$$r(\theta) = C_x + R\sin(\theta) \tag{19}$$

$$c(\theta) = C_y + R\cos(\theta) \tag{20}$$

In Eq. (19) and Eq. (20), the radius R is a predefined constant and $\theta \in (0, 2\pi)$. To achieve optimal performance, the hyperparameters α, β, and γ are modified. In Eq. (2), α controls the contour's elasticity and β controls the rigidity or smoothness. γ is the explicit time-stepping parameter that influences the model's convergence speed. These adjustments enable a balanced trade-off between the internal and external energies, which allows the model to dynamically adjust and encapsulate the ulcer contours and effectively detect the ulcer boundaries. From this segmented contour, bounding box coordinates are extracted and validated to ensure adherence to image dimensions and optimise subsequent stages.

4.5.2. Sobel Edge Detection (SED)

The Sobel operator is utilized within the area defined by the iACM-generated bounding box to accurately detect and highlight prominent edges. This enhances the visual clarity of the ulcer boundaries and aids in extracting precise contours. Following this, a gradient magnitude image is generated to capture cohesive edge information within the ulcer region.

4.5.3. Binary Mask Creation

The gradient magnitude image undergoes thresholding to create a binary mask emphasising the important edges. This produces a simplified visualisation of the ulcer region for the next stage. The resulting binary mask is presented in Figure 5.

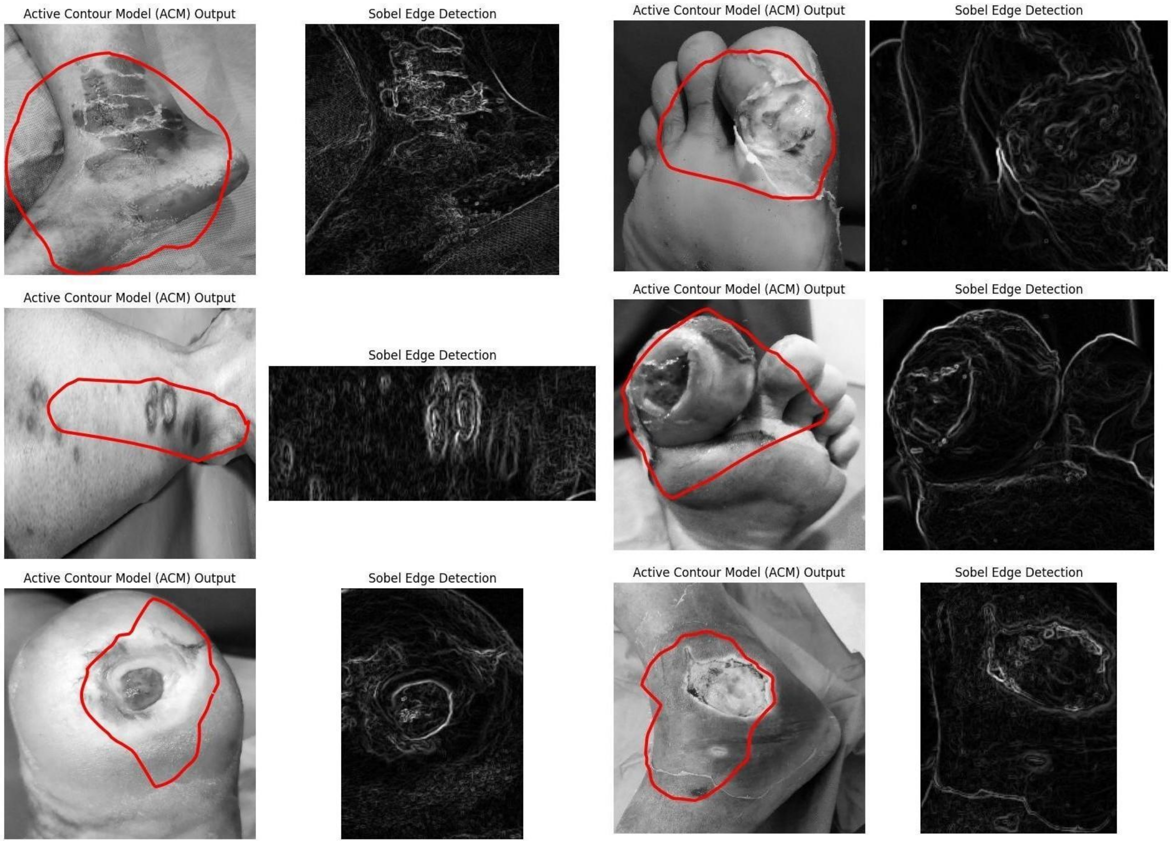

Active Contour Model (ACM) Output
Sobel Edge Detection
Active Contour Model (ACM) Output
Sobel Edge Detection
Active Contour Model (ACM) Output
Sobel Edge Detection
Active Contour Model (ACM) Output
Sobel Edge Detection
Active Contour Model (ACM) Output
Sobel Edge Detection
Active Contour Model (ACM) Output
Sobel Edge Detection

Figure 5: Simplified Visualization of Ulcer Region after applying iACM and SED

4.5.4. Largest Contour Extraction

First, we create a refined binary mask and then extract contours from it. We analyze these contours to select the one with the largest area, which represents the ulcer region. This is the final step in accurately segmenting the ulcer. The extraction process isolates the most significant contour representing the ulcer region while disregarding thin lines and smaller objects. Subsequently, a bounding box is generated around the isolated ulcer to ensure it aligns with the image dimensions, as illustrated in Figure 6. The coordinates of the bounding box for each image are then saved in XML VOC format. The resulting XML file is utilized in the subsequent classification stage to identify the Region of Interest (ROI) for each image and grade the ulcers based on the Wagner grading scale.

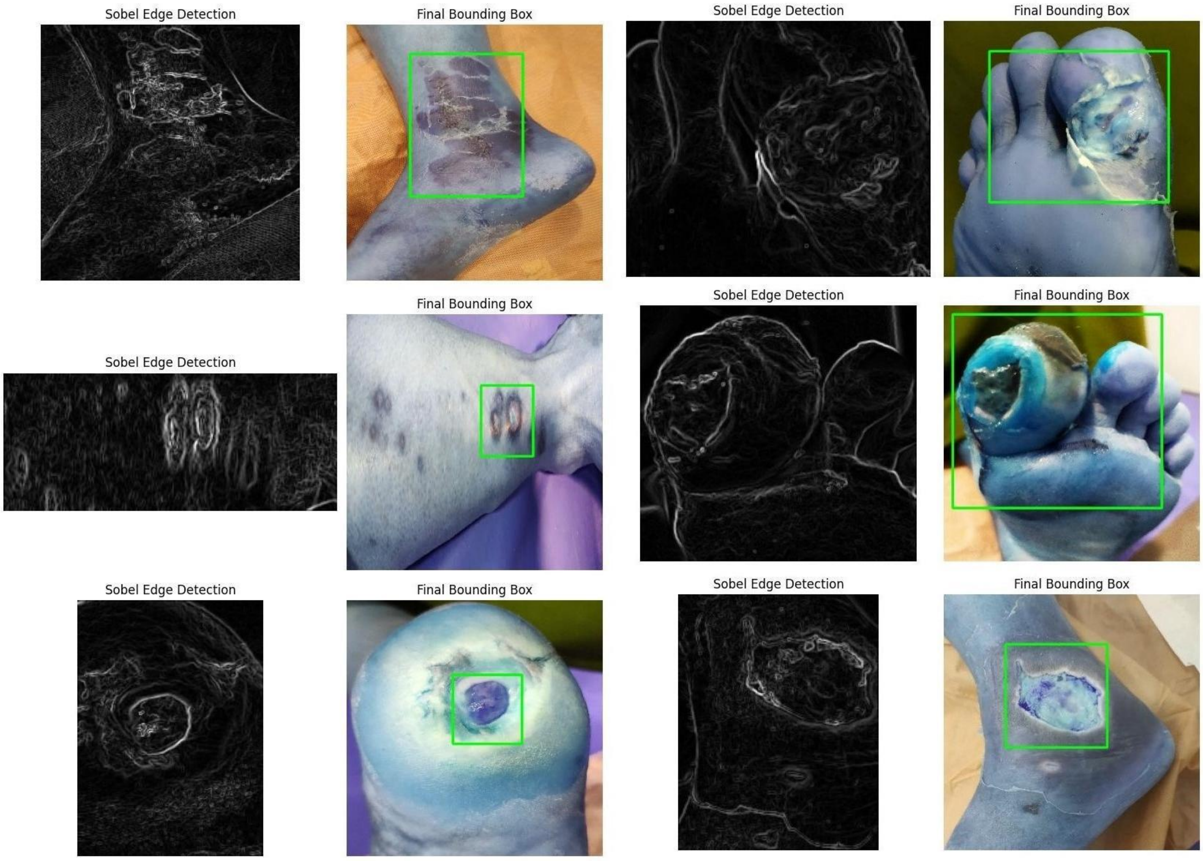


Figure 6: Segmented Bounding Box

4.6. Classification using Wagner's Grading System

The classification tasks are performed using MobileNetV3-Small, pre-trained on ImageNet and fine-tuned for resource-constrained mobile phone CPUs through a platform-aware network architecture search and the NetAdapt algorithm. MobileNetV3-Small is chosen over MobileNetV3-Large because it is lightweight and provides promising results comparable to other deep learning architectures. The dataset contains images with corresponding XML files and bounding box information in VOC format. The framework of the classification architecture is presented in Figure 7.

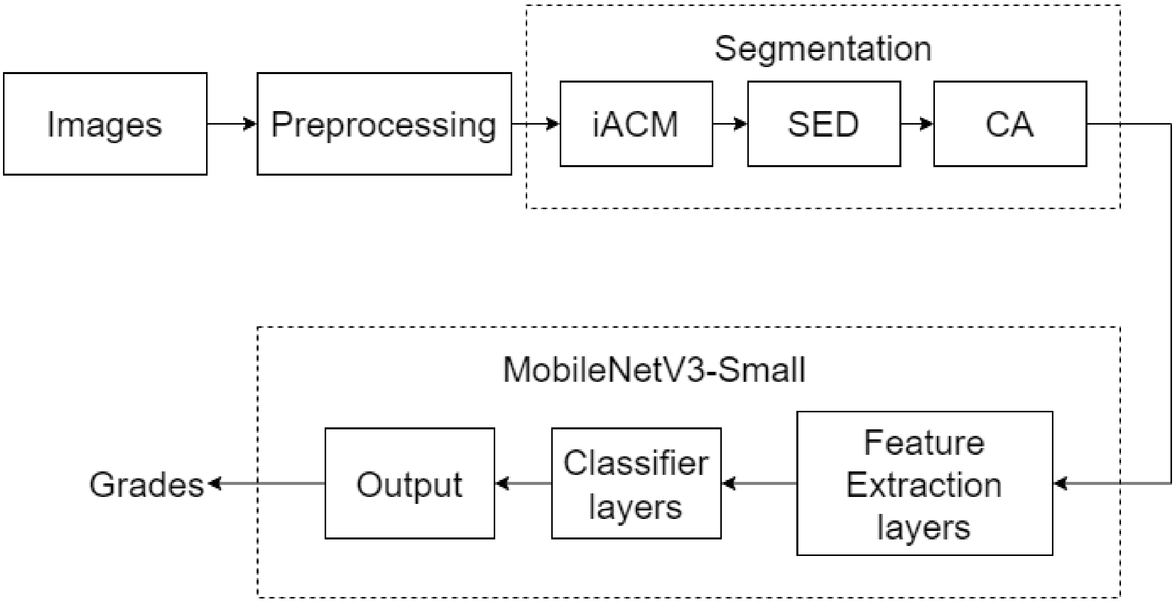


Figure 7: Framework of the proposed segmentation (iACM+SED+CA) method and classification architecture

In the next section, the proposed method is evaluated using standard metrics to assess its ability to classify the ulcers precisely into their respective grades.

5. Experiments, Results and Analysis

In this project, we used Python version 3.11.4 along with various built-in libraries. The Scikit-image library, an open-source image processing library, was used for tasks such as image preprocessing, morphological operations, and edge detection. We imported the Active Contour Model from OpenCV (Open Computer Vision), a real-time computer vision and machine learning library. For implementing the MobileNetV3-Small architecture, we utilized PyTorch, an open-source machine learning framework based on the Torch library.

The dataset consists of 923 image samples, which were augmented to generate 4987 samples. The number of samples in each grade in the augmented dataset is depicted in Table 3. The dataset is split into a training set (80% of the data, 3990 samples) and a testing set (20% of the data, 997 samples).

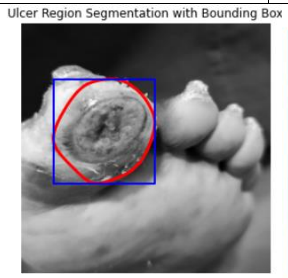
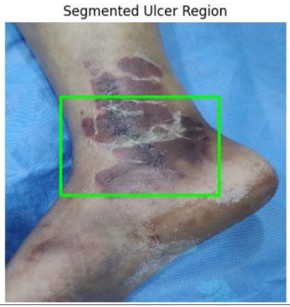
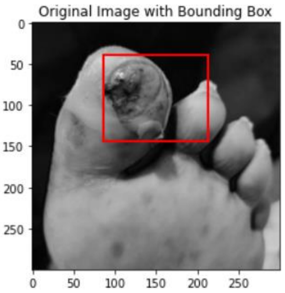
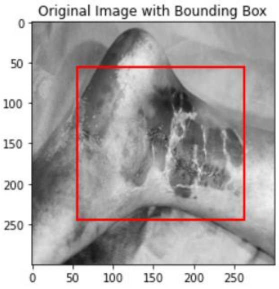
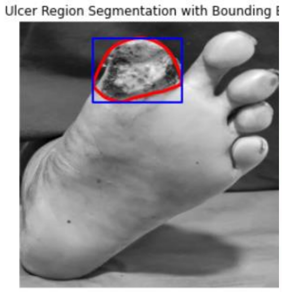
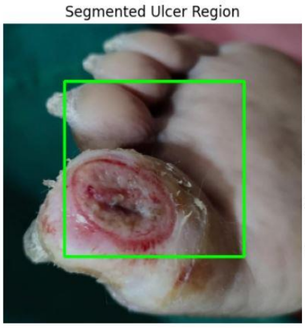
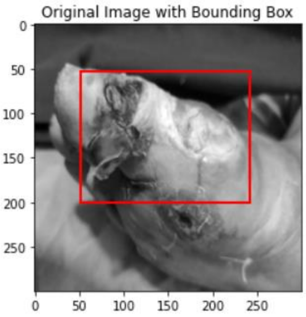
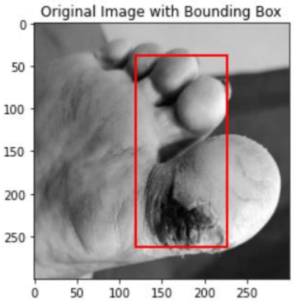
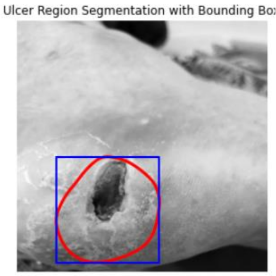
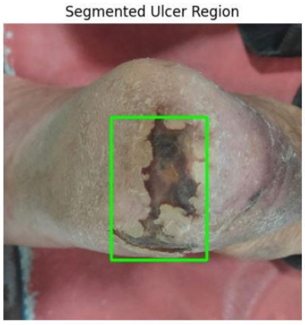
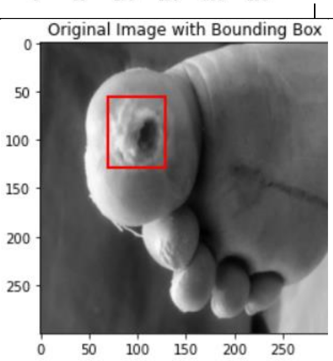
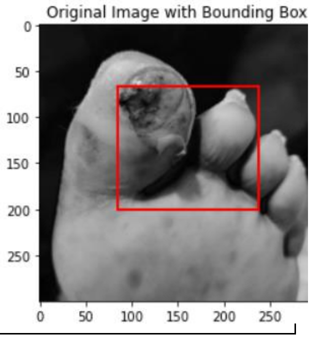
Table 3: Number of DFU images in different grades in the augmented dataset

Ulcer Grades	Description	Number of Images
Grade-0	Pre foot ulceration	821
Grade-1	Partial foot ulceration	832
Grade-2	Deep ulcer to bone, ligament, tendon, or joint	825
Grade-3	Deep foot ulcer with osteomyelitis	836
Grade-4	Ischemic -foot gangrene	840
Grade-5	Entire foot gangrene	833

Various operations were performed for quality enhancement, including one iteration of closing and two iterations of opening using a 3x3 kernel. The images were resized to 300x300 pixels for generalization. The initial contour of the iACM was set with a radius (R) of 150, and specific hyperparameters (alpha, beta, and gamma) were set to 0.06, 0.1, and 0.001, respectively. Edge detection was performed using the Sobel operator with a 3x3 kernel.

To identify the outlines of objects, the cv2.findContours() function is used to extract contours from the binary mask. The area enclosed by each contour was obtained using the cv2.contourArea() function, and the contour with the largest area was selected as the ulcer. XML files containing RoI coordinates in adherence to the VOC format were generated from the segmentation stage. The images were converted to PyTorch tensors and normalized using mean values of [0.485, 0.456, 0.406] and standard deviations of [0.229, 0.224, 0.225] before being used as inputs to the classifier.

The MobileNetV3-Small with 17 layers was deployed for classification, and the final classifier layer was modified according to the number of classes in the dataset. The classifier was trained across 5 epochs using the Adam optimizer with a learning rate (LR) hyperparameter of 0.001. CrossEntropyLoss was used as the loss function, and a training batch size of 32 was selected to ensure efficient memory utilization. The experiments show that the images segmented with the proposed iACM+SED+CA method are accurate and clear.

iACM+SED+CA	**ACM**	**Fuzzy C-Means Clustering**	**GVF**
		 Original Image with Bounding Box 0 50 100 150 200 250 0 50 100 150 200 250	 Original Image with Bounding Box 0 50 100 150 200 250 0 50 100 150 200 250
		 Original Image with Bounding Box 0 50 100 150 200 250 0 50 100 150 200 250	 Original Image with Bounding Box 0 50 100 150 200 250 0 50 100 150 200 250
		 Original Image with Bounding Box 0 50 100 150 200 250 0 50 100 150 200 250	 Original Image with Bounding Box 0 50 100 150 200 250 0 50 100 150 200 250

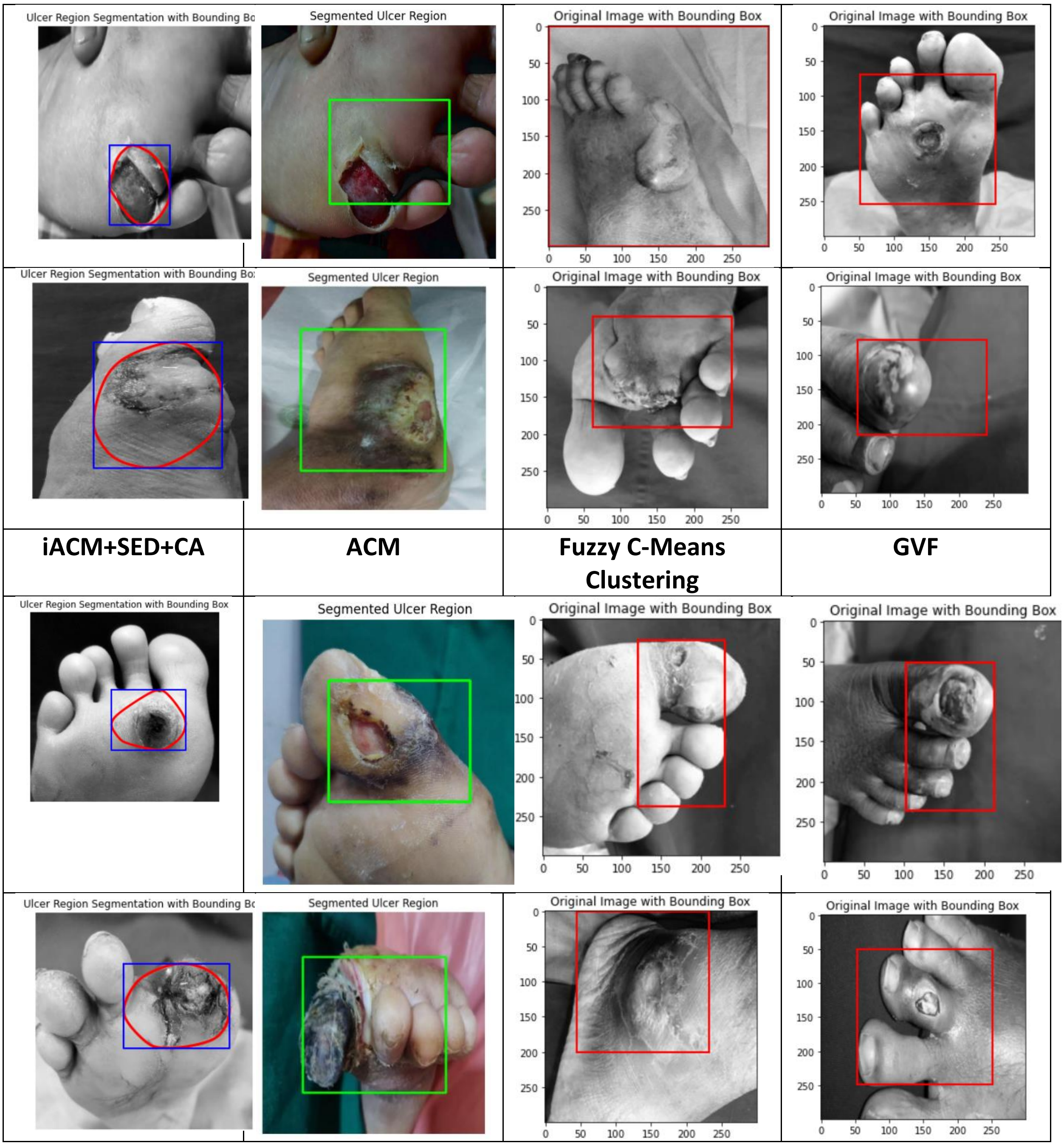

Ulcer Region Segmentation with Bounding Bo
Segmented Ulcer Region
Original Image with Bounding Box
Original Image with Bounding Box
Ulcer Region Segmentation with Bounding Bo:
Segmented Ulcer Region
Original Image with Bounding Box
Original Image with Bounding Box
iACM+SED+CA
ACM
Fuzzy C-Means Clustering
GVF
Ulcer Region Segmentation with Bounding Box
Segmented Ulcer Region
Original Image with Bounding Box
Original Image with Bounding Box
Ulcer Region Segmentation with Bounding Bc
Segmented Ulcer Region
Original Image with Bounding Box
Original Image with Bounding Box
0
50
100
150
200
250

Figure 8: Comparative analysis of the proposed (IACM + SED+ CA) method with other segmentation techniques

To assess the performance of the iACM + SED+ CA method and the MobileNetV3-Small classifier, four metrics are considered: accuracy, precision, recall, and F1-score. To compare, all the images are segmented using the proposed segmentation method and four other existing segmentation methods [29, 38, 39, 40]. The region of interest (RoI) extracted from these segmentation methods [29, 38, 39, 40] is then used for classification with the

MobileNetV3-Small. Table 4 presents the quantitative metrics obtained from the classification with the RoI extracted from each segmentation method.

Table 4: Comparative evaluation of the performance metrics with the proposed iACM+SED+CA method and the other segmentation methods

Evaluation Metrics	**ACM with MobileNetV3 -Small**	**GVF with MobileNetV3 -Small**	**FCM with MobileNetV3 -Small**	**ResNet18 with MobileNetV3 -Small**	**iACM+ SED+ CA with MobileNetV3 -Small**
Accuracy	0.8211	0.8767	0.6475	0.8457	0.8948
Precision	0.8615	0.8914	0.7688	0.8656	0.9189
Recall	0.8212	0.8767	0.6474	0.8457	0.8948
F1-Score	0.8066	0.8734	0.6138	0.8454	0.8983

The proposed iACM+SED+CA method and the MobileNetV3-Small classifier achieved the highest accuracy of 0.8948, precision of 0.9189, recall of 0.8948, and F1-Score of 0.8983. The experimental results demonstrate that the RoI extracted by the iACM+SED+CA method significantly enhances classification with MobileNetV3-Small compared to other segmentation methods. Furthermore, Figure 8 illustrates that the proposed method effectively delineated the foot ulcer boundary compared to other commonly used segmentation techniques. Therefore, it can be inferred that the iACM+SED+CA method delivers high performance.

6. Conclusion and Future Scope

The proposed method utilises an improved Active Contour Model (iACM), Sobel Edge Detection and Contour Analysis to precisely isolate the ulcer regions from foot ulcer images, followed by classification using MobileNetV3-Small, favoured by limited resource devices, to provide a lightweight yet robust solution for real-time foot ulcer grading as per the Wagner grade classification system. A dataset comprising 923 images with 6 severity grades is collected to evaluate the method. The proposed approach automates the segmentation process using a novel segmentation algorithm that effectively isolates the ulcer region without relying on built-in segmentation tools. Since the Active Contour Model (ACM) is sensitive to noise, binarisation followed by closing and opening operations were applied for simplification and noise removal, respectively. Image moments dynamically initialise the contour without relying on manual initialisation. Hyper-parameter optimisation is performed to improve the overall precision of the ACM. The isolated region is further refined by Sobel Edge Detection (SED), followed by Contour Analysis (CA) to detect the largest contour and extract the ulcer region. Then, the segmented regions of interest (RoI) are used by MobileNetV3-Small to classify the foot ulcers into different severity grades (0-5).

The comparative analysis shows that the proposed method provides promising results in segmenting the ulcer region from DFU images, allowing the classifier to grade the foot ulcers to their respective severity grades accurately. The quantitative evaluation shows that the proposed algorithm more effectively detects and isolates the ulcer region and is not inferior to the previously mentioned state-of-the-art methods. Overall, the classification and prediction of foot ulcers using a low-constraint classification architecture provides real-time solutions, thus promoting timely clinical decisions.

Future work includes incorporating shadow removal techniques to handle images with poor illumination and integrating the proposed classification and prediction method in a mobile environment to offer customised treatment plans, prompt clinical actions, and identify high-risk patients.

Reference

1. University of California, San Francisco, "Diabetic Foot Ulcers," UCSF Vascular and Endovascular Surgery. [Online]. Available: https://vascularsurgery.ucsf.edu/conditions--procedures/diabetic-foot-ulcers.aspx. [Accessed: April 21, 2024].
2. S. Das, P. Roy, and A. K. Mishra, “Deep Learning Techniques Dealing with Diabetes Mellitus: A Comprehensive Study,” Studies in computational intelligence, pp. 295–323, Jan. 2021, doi: 10.1007/978-981-15-9735-0_15.
3. P. Saeedi, et al., “Global and Regional Diabetes Prevalence Estimates for 2019 and Projections for 2030 and 2045: Results from the International Diabetes Federation Diabetes Atlas, 9th Edition,” Diabetes Research and Clinical Practice, vol. 157, no. 157, pp. 295–323, Sep. 2019, doi: 10.1016/j.diabres.2019.107843.
4. T. I. Oliver and M. Mutluoglu, “Diabetic Foot Ulcer,” PubMed, Aug. 08, 2023. [Online]. Available: https://www.ncbi.nlm.nih.gov/books/NBK537328/
5. P. Zhang, J. Lu, Y. Jing, S. Tang, D. Zhu, and Y. Bi, “Global epidemiology of diabetic foot ulceration: a systematic review and meta-analysis,” Annals of Medicine, vol. 49, no. 2, pp. 106–116, Nov. 2016, doi: 10.1080/07853890.2016.1231932.
6. G. Akkus and M. Sert, “Diabetic foot ulcers: A devastating complication of diabetes mellitus continues non-stop in spite of new medical treatment modalities,” World Journal of Diabetes, vol. 13, no. 12, pp. 1106–1121, Dec. 2022, doi: 10.4239/wjd.v13.i12.1106.
7. Z. J. Lo, N. K. Surendra, A. Saxena, and J. Car, “Clinical and economic burden of diabetic foot ulcers: A 5-year longitudinal multi-ethnic cohort study from the tropics,” International Wound Journal, vol. 18, no. 3, pp. 375–386, Jan. 2021, doi: 10.1111/iwj.13540.
8. M. Goyal, A. Oakley, P. Bansal, D. Dancey, and M. H. Yap, “Skin Lesion Segmentation in Dermoscopic Images With Ensemble Deep Learning Methods,” IEEE Access, vol. 8, pp. 4171–4181, 2020, doi: 10.1109/access.2019.2960504.
9. L. Wang, P. C. Pedersen, E. Agu, D. M. Strong, and B. Tulu, “Area Determination of Diabetic Foot Ulcer Images Using a Cascaded Two-Stage SVM-Based Classification,” IEEE Transactions on Biomedical Engineering, vol. 64, no. 9, pp. 2098–2109, Sep. 2017, doi: 10.1109/tbme.2016.2632522.

10. Dr. M. Mehraj, “A review of Wagner classification and current concepts in management of diabetic foot,” International Journal of Orthopaedics Sciences, vol. 4, no. 1n, pp. 933–935, Jan. 2018, doi: 10.22271/ortho.2018.v4.i1n.133.
11. F. W. Wagner, “The diabetic foot” Orthopaedics, vol. 10, no. 1, pp. 163-172, 1987.
12. A. J. M. Boulton, et al., Diagnosis and Management of Diabetic Foot Complications. Arlington (VA): American Diabetes Association, 2018. [Online]. Available: https://www.ncbi.nlm.nih.gov/books/NBK538977/
13. A. Han, et al., “Deep Learning Methods for Real-time Detection and Analysis of Wagner Ulcer Classification System,” in 2022 International Conference on Computer Applications Technology (CCAT), Guangzhou, China, Jul. 2022, pp. 11-21, doi: 10.1109/ccat56798.2022.00010.
14. A. Han, et al., "Application of Refinements on Faster-RCNN in Automatic Screening of Diabetic Foot Wagner Grades," Acta Medica Mediterranea, vol. 36, pp. 661-666, 2020, doi: 10.19193/0393-6384_2020_1_104.
15. E. Pasero and C. Castagneri, “Application of an automatic ulcer segmentation algorithm,” 2017 IEEE 3rd International Forum on Research and Technologies for Society and Industry (RTSI), Sep. 2017, doi: 10.1109/rtsi.2017.8065954.
16. A. Oliveira, A. Britto de Carvalho, and D. Dantas, “Faster R-CNN Approach for Diabetic Foot Ulcer Detection,” Proceedings of the 16th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, 2021, doi: 10.5220/0010255506770684.
17. P. Ge, Y. Chen, et al., “An active contour model based on Jeffreys divergence and clustering technology for image segmentation,” Journal of visual communication and image representation, vol. 99, Mar. 2024, Art. no. 104069, doi: 10.1016/j.jvcir.2024.104069.
18. K. Cheng, T. Xiao, Q. Chen, and Y. Wang, “Image segmentation using active contours with modified convolutional virtual electric field external force with an edge-stopping function,” PLOS ONE, vol. 15, no. 3, Mar. 2020, Art. no. e0230581, doi: 10.1371/journal.pone.0230581.
19. L. Khrissi, N. E. Akkad, H. Satori, and K. Satori, “Image Segmentation Based on K-means and Genetic Algorithms,” Advances in intelligent systems and computing, pp. 489–497, Jan. 2020, doi: 10.1007/978-981-15-0947-6_46.

20. S. Back et al., "Robust Skin Disease Classification by Distilling Deep Neural Network Ensemble for the Mobile Diagnosis of Herpes Zoster," in IEEE Access, vol. 9, pp. 20156-20169, Jan. 2021, doi: 10.1109/ACCESS.2021.3054403.
21. J. Canny, “A Computational Approach to Edge Detection,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-8, no. 6, pp. 679–698, Nov 1986, doi: 10.1109/tpami.1986.4767851.
22. M. Kass, A. Witkin, and D. Terzopoulos, “Snakes: Active contour models,” International Journal of Computer Vision, vol. 1, no. 4, pp. 321–331, Jan. 1988, doi: 10.1007/bf00133570.
23. Sobel and G. M. Feldman, "An Isotropic 3×3 image gradient operator," Semantic Scholar, 1990. [Online]. Available: https://api.semanticscholar.org/CorpusID:59909525.
24. A. Heras-Tang, D. Valdes-Santiago, Á. M. León-Mecías, M. L. Baguer Díaz-Romañach, J. A. Mesejo-Chiong, and C. Cabal-Mirabal, “Diabetic foot ulcer segmentation using logistic regression, DBSCAN clustering and mathematical morphology operators,” ELCVIA Electronic Letters on Computer Vision and Image Analysis, vol. 21, no. 2, pp. 22–39, Sep. 2022, doi: 10.5565/rev/elcvia.1413.
25. C. Wang et al., “Fully automatic wound segmentation with deep convolutional neural networks,” Scientific Reports, vol. 10, no. 1, Dec. 2020, doi: 10.1038/s41598-020-78799-w.
26. B. K S, S. Sabut, and N. D K, “Efficient Detection and Classification of Diabetic Foot Ulcer Tissue using PSO Technique,” International Journal of Engineering & Technology, vol. 7, no. 3.12, pp. 1006, Jul. 2018, doi: 10.14419/ijet.v7i3.12.17622.
27. R. Niri, H. Douzi, Y. Lucas, and S. Treuillet, "A Superpixel-Wise Fully Convolutional Neural Network Approach for Diabetic Foot Ulcer Tissue Classification," in Pattern Recognition. ICPR International Workshops and Challenges, Feb. 2021, pp. 308–320, doi: 10.1007/978-3-030-68763-2_23.
28. B. C. Motta, M. P. Marques, M.P. Marques, G.A. Guimarães, R.U. Ferreira and S.S.R.F. Rosa, “The evaluation of the healing process of diabetic foot wounds using image segmentation and neural networks classification,” International journal of biomedical engineering and technology, vol. 38, no. 2, pp. 179–179, Feb. 2022, doi: 10.1504/ijbet.2022.120869.

29. V. Rajathi, A. Chinnasamy, and P. Selvakumari, “DUTC Net: A novel deep ulcer tissue classification network with stage prediction and treatment plan recommendation,” Biomedical signal processing and control, vol. 90, Apr 2024, Art. no. 105855, doi: 10.1016/j.bspc.2023.105855.
30. D. Wang, X. Dang, W. Liu, and Y. Wang, “Image segmentation using active contours with image structure adaptive gradient vector flow external force,” Frontiers in applied mathematics and statistics, vol. 9, p. 1271296, Oct. 2023, doi: 10.3389/fams.2023.1271296.
31. X.-X. Yin et al., “Automatic breast tissue segmentation in MRIs with morphology snake and deep denoiser training via extended Stein’s unbiased risk estimator,” Health information science and systems, vol. 9, Apr. 2021, Art. no. 16, doi: 10.1007/s13755-021-00143-x.
32. Z. Peng, L. Wang, L. Tong, H. Zou, D. Liu, and C. Zhang, “Multi-threshold image segmentation of 2D OTSU inland ships based on improved genetic algorithm,” PloS one, vol. 18, no. 8, Aug. 2023, Art. no. e0290750, doi: 10.1371/journal.pone.0290750.
33. J. Cervante Canales, J. Cervante Canales, F. García-Lamont, A. Yee Rendón, J. S. Ruiz Castilla, and L. Rodríguez-Mazahua, “Optimal segmentation of image datasets by genetic algorithms using color spaces,” Expert systems with applications, vol. 238, Mar. 2024, Art. no. 121950, doi: 10.1016/j.eswa.2023.121950.
34. C. Yu, Y. Wang, C. Tang, W. Feng, and J. Lv, “EU-Net: Automatic U-Net neural architecture search with differential evolutionary algorithm for medical image segmentation,” Computers in biology and medicine, vol. 167, Dec. 2023, Art. no. 107579, doi: 10.1016/j.compbiomed.2023.107579.
35. A. M. P. Boelens and H. A. Tchelepi, “QuantImPy: Minkowski functionals and functions with Python,” SoftwareX, vol. 16, Dec. 2021, Art. no. 100823, doi: 10.1016/j.softx.2021.100823.
36. S. Bhutada, N. Yashwanth, P. Dheeraj, and K. Shekar, “Opening and closing in morphological image processing,” World Journal of Advanced Research and Reviews, vol. 14, no. 3, pp. 687–695, Jun. 2022, doi: 10.30574/wjarr.2022.14.3.0576.
37. K. Bapat, "Find Center of Blob (Centroid) Using OpenCV (C++/Python)," LearnOpenCV, 2018. [Online]. Available: https://learnopencv.com/find-

center-of-blob-centroid-using-opencv-cpp-python/. [Accessed: March 27, 2024].

38. A. Maryani, M. C. Anwar, and B. Abimanyu, “Accuracy of Prostate Volume Measurement on 2d Transabdominal USG Modality Using the Gradient Vector Flow (GVF) Segmentation Application Measurement Technique,” International journal of social health, vol. 2, no. 10, pp. 725–733, Nov. 2023, doi: 10.58860/ijsh.v2i10.107.
39. W. Wiharto and E. Suryani, “The Comparison of Clustering Algorithms K-Means and Fuzzy C-Means for Segmentation Retinal Blood Vessels,” Acta Informatica Medica, vol. 28, no. 1, pp. 42-47, Mar. 2020, doi: 10.5455/aim.2020.28.42-47.
40. W. Gómez-Flores and W. Coelho de Albuquerque Pereira, “A comparative study of pre-trained convolutional neural networks for semantic segmentation of breast tumours in ultrasound,” Computers in Biology and Medicine, vol. 126, Nov. 2020, Art. no. 104036, doi: 10.1016/j.compbiomed.2020.104036.

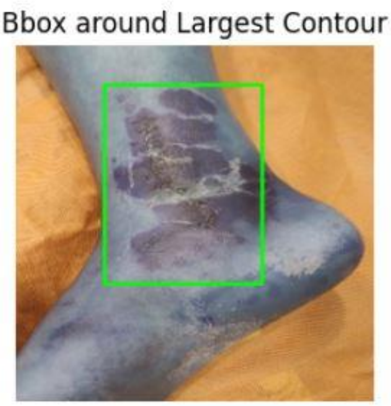
Appendix

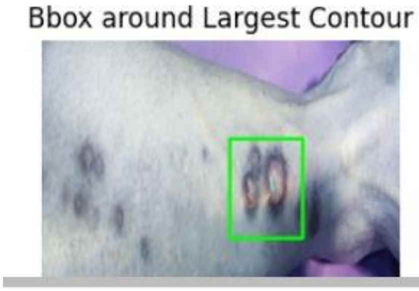
DFU Segmentation and Classification approach using Improved ACM-based Sobel + Contour Analysis

```
# Importing libraries
import os
import cv2
import numpy as np
import matplotlib.pyplot as plt
from skimage.segmentation import active_contour

# Function to perform preprocessing and ulcer region segmentation
def preprocess_and_segment(image_path):
    # Load the image
    image = cv2.imread(image_path)

    # Resize the image
    image = cv2.resize(image, (300, 300))

    # Convert the image to grayscale
    gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)

    # Texture analysis for rough areas
    texture = cv2.Canny(gray, 50, 150)

    # Thresholding to identify ulcer regions
    _, binary = cv2.threshold(texture, 80, 255, cv2.THRESH_BINARY)

    # Apply morphological operations to refine the binary mask
    kernel = np.ones((3, 3), np.uint8)
    # Closing
    binary = cv2.morphologyEx(binary, cv2.MORPH_CLOSE, kernel)
    # Opening
    binary = cv2.erode(binary, kernel, iterations=2)
    binary = cv2.dilate(binary, kernel, iterations=2)
```

```
# Initialize snake contour based on image moments
moments = cv2.moments(binary)
c_x = int(moments["m01"] / moments["m00"])
c_y = int(moments["m10"] / moments["m00"])
s = np.linspace(0, 2*np.pi, 100)
r = c_x + 150 * np.sin(s)
c = c_y + 150 * np.cos(s)
init = np.array([r, c]).T

# Run active contour (snake algorithm) to segment ulcer regions
snake = active_contour(binary, init, alpha=0.06, beta=0.1, gamma=0.001)

# Extract coordinates for bounding box
min_x, min_y = np.min(snake, axis=0)
max_x, max_y = np.max(snake, axis=0)

# Ensure bounding box coordinates are within image dimensions
min_x = max(min_x, 0)
min_y = max(min_y, 0)
max_x = min(max_x, image.shape[0])
max_y = min(max_y, image.shape[1])

# Plot the segmented ulcer region with bounding box
# plt.figure(figsize=(10, 5))
# plt.subplot(1, 2, 1)
# plt.imshow(gray, cmap='gray')
# plt.plot(snake[:, 1], snake[:, 0], '-r', lw=3)
# plt.plot([min_y, max_y, max_y, min_y, min_y], [min_x, min_x, max_x, max_x, min_x], '-b', lw=2)
# plt.title('Ulcer Region Segmentation with Bounding Box')
# plt.axis('off')

# Sobel edge detection within the bounding box
bbox_image = gray[int(min_x):int(max_x), int(min_y):int(max_y)]
```

```
    if bbox_image.size > 0:
        sobel_x = cv2.Sobel(bbox_image, cv2.CV_64F, 1, 0, ksize=3)
        sobel_y = cv2.Sobel(bbox_image, cv2.CV_64F, 0, 1, ksize=3)
        gradient_magnitude = np.sqrt(sobel_x ** 2 + sobel_y ** 2)

        # Thresholding for binary masking
        _, dark_blob_mask = cv2.threshold(gradient_magnitude, 50, 255, cv2.THRESH_BINARY)

        # Find contours of dark pixel blobs
        contours, _ = cv2.findContours(dark_blob_mask.astype(np.uint8), cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)

        # Find contour with the largest area within the bounding box
        if contours:
            max_contour = max(contours, key=cv2.contourArea)
            x, y, w, h = cv2.boundingRect(max_contour)

            # Ensure the contour is not too close to the bounding box border
            margin = 5
            x = max(0, x - margin)
            y = max(0, y - margin)
            w = min(w + 2 * margin, dark_blob_mask.shape[1] - x)
            h = min(h + 2 * margin, dark_blob_mask.shape[0] - y)

            # Draw the bounding box on the original image
            bbox_img = cv2.rectangle(image.copy(), (int(min_y) + x, int(min_x) + y), (int(min_y) + x + w, int(min_x) + y + h), (0, 255, 0), 2)

            # Plot the processed mask and bounding box
            plt.subplot(1, 2, 2)
            plt.imshow(bbox_img)
            plt.title('Bbox around Largest Contour')
            plt.axis('off')
            plt.show()

            # Save segmented region in VOC format
            with open(output_path, 'w') as f:

f.write(f'<annotation>\n\t<folder>output</folder>\n\t<filename>{os.path.basename(image_path)}</filename>\n')

f.write('\t<source>\n\t\t<database>Unknown</database>\n\t</source>\n')

f.write(f'\t<size>\n\t\t<width>{image.shape[1]}</width>\n\t\t<height>{image.shape[0]}</height>\n\t\t<depth>{image.shape[2]}</depth>\n\t</size>\n')
                f.write('\t<segmented>0</segmented>\n')

f.write(f'\t<object>\n\t\t<name>ulcer</name>\n\t\t<pose>Unspecified</pose>\n')
                f.write('\t\t<truncated>0</truncated>\n\t\t<difficult>0</difficult>\n')
                f.write(f'\t\t<bndbox>\n\t\t\t<xmin>{int(min_y) + x}</xmin>\n\t\t\t<ymin>{int(min_x) + y}</ymin>\n')
                f.write(f'\t\t\t<xmax>{int(min_y) + x + w}</xmax>\n\t\t\t<ymax>{int(min_x) + y + h}</ymax>\n\t\t</bndbox>\n\t</object>\n')
                f.write('</annotation>')

    plt.show()

# Path to the data folder
data_folder = "augmented_data"

# Iterate over subfolders in the data folder
for subfolder in os.listdir(data_folder):
    subfolder_path = os.path.join(data_folder, subfolder)
    if os.path.isdir(subfolder_path):
        output_folder = os.path.join("steps2", subfolder)
        os.makedirs(output_folder, exist_ok=True)
        # Iterate over images in the subfolder
        for filename in os.listdir(subfolder_path):
            if filename.endswith(".jpg") or filename.endswith(".jpeg") or
filename.endswith(".png"):
                image_path = os.path.join(subfolder_path, filename)
                output_path = os.path.join(output_folder,
os.path.splitext(filename)[0] + ".xml")
                preprocess_and_segment(image_path)
```





```
import torch
import torch.nn as nn
import torch.optim as optim
import torchvision.transforms as transforms
import torchvision.models as models
from sklearn.metrics import accuracy_score, precision_score, recall_score,
f1_score
from xml.etree import ElementTree as ET
from torch.utils.data import Dataset, DataLoader
from torchvision.models import mobilenet_v3_small
```

```
# Parsing XML files and extracting ROI
def parse_voc_xml(xml_file):
    tree = ET.parse(xml_file)
    root = tree.getroot()
    objects = root.findall('object')
    rois = []
    for obj in objects:
        bndbox = obj.find('bndbox')
        xmin = int(bndbox.find('xmin').text)
        ymin = int(bndbox.find('ymin').text)
        xmax = int(bndbox.find('xmax').text)
        ymax = int(bndbox.find('ymax').text)
        rois.append((xmin, ymin, xmax, ymax))
    return rois

# Creating Ulcer dataset with images and ROIs
class UlcerDataset(Dataset):
    def __init__(self, data_folder, output_folder, transform=None):
        self.data_folder = data_folder
        self.output_folder = output_folder
        self.transform = transform
        self.images = []
        self.labels = []
        self.class_to_index = {}
        for i, subfolder in enumerate(sorted(os.listdir(data_folder))):  # Assign unique index for each class
            self.class_to_index[subfolder] = i
            subfolder_path = os.path.join(data_folder, subfolder)
            output_subfolder = os.path.join(output_folder, subfolder)
            if os.path.isdir(subfolder_path) and os.path.isdir(output_subfolder):
                for filename in os.listdir(subfolder_path):
                    if filename.endswith(".jpg") or filename.endswith(".jpeg") or filename.endswith(".png"):
                        image_path = os.path.join(subfolder_path, filename)
                        xml_path = os.path.join(output_subfolder,
os.path.splitext(filename)[0] + ".xml")
                        if os.path.exists(xml_path):
                            image = cv2.imread(image_path)
                            image_resized = cv2.resize(image, (300, 300))
                            rois = parse_voc_xml(xml_path)
                            self.images.append(image_resized)
                            self.labels.append(subfolder)  # Store class name as label

    def __len__(self):
        return len(self.images)

    def __getitem__(self, idx):
        image = self.images[idx]
        label = self.labels[idx]
        if self.transform:
            image = self.transform(image)
        return image, self.class_to_index[label]  # Use index instead of label

# Transformations
data_transforms = transforms.Compose([
    transforms.ToTensor(),
    transforms.Normalize(mean=[0.485, 0.456, 0.406], std=[0.229, 0.224,
0.225])
])

# Data loading from folders
data_folder = "aug_data"
output_folder = "output"
dataset = UlcerDataset(data_folder, output_folder,
transform=data_transforms)

# Train and test split
train_size = int(0.75 * len(dataset))
test_size = len(dataset) - train_size
```

```
train_dataset, test_dataset = torch.utils.data.random_split(dataset,
[train_size, test_size])

# Creating data loaders
train_loader = DataLoader(train_dataset, batch_size=32, shuffle=True)
test_loader = DataLoader(test_dataset, batch_size=32)

# Initializing MobileNetV3
model = mobilenet_v3_small(pretrained=True, progress=True)

# Modifying the classifier head
num_ftrs = model.classifier[3].in_features
model.classifier[3] = nn.Linear(num_ftrs, len(dataset.class_to_index))

# Loss function and optimizer
criterion = nn.CrossEntropyLoss()
optimizer = optim.Adam(model.parameters(), lr=0.001)

# Model training
model.train()
for epoch in range(5):
    running_loss = 0.0
    for inputs, labels in train_loader:
        optimizer.zero_grad()
        outputs = model(inputs)
        loss = criterion(outputs, labels)
        loss.backward()
        optimizer.step()
        running_loss += loss.item() * inputs.size(0)
    epoch_loss = running_loss / len(train_loader.dataset)
    print(f"Epoch {epoch+1}/{5}, Loss: {epoch_loss:.4f}")
```

```
Epoch 1/5, Loss: 0.7624

Epoch 2/5, Loss: 0.2852

Epoch 3/5, Loss: 0.1709
Epoch 4/5, Loss: 0.1158

Epoch 5/5, Loss: 0.0776
```

```
# Model evaluation
model.eval()
all_labels = []
all_preds = []
with torch.no_grad():
    for inputs, labels in test_loader:
        outputs = model(inputs)
        _, preds = torch.max(outputs, 1)
        all_labels.extend(labels.numpy())
        all_preds.extend(preds.numpy())

# Calculate precision, recall, and F1-scores
precision = precision_score(all_labels, all_preds, average='weighted',
zero_division=1)
recall = recall_score(all_labels, all_preds, average='weighted', zero_division=1)
f1 = f1_score(all_labels, all_preds, average='weighted', zero_division=1)
accuracy = accuracy_score(all_labels, all_preds)

print("Accuracy:", accuracy)
print("Precision:", precision)
print("Recall:", recall)
print("F1-score:", f1)
```

```
Accuracy: 0.8948220064724919

Precision: 0.9188810433075294

Recall: 0.8948220064724919

F1-score: 0.8982661491335232
```

…………………THE END………………