Segmentation and Classification of skin lesion images using deep learning based techniques

A report submitted in partial fulfilment of the requirements for the award of the degree of

Bachelor of Technology
in
Electronics and Communication Engineering

by

Devvjiit Bhuyan	**ECB19050**
Md Noman	**ECB19053**
Gulshan Kumar	**ECB19057**
Puralasetty Sumanth	**ECB19060**

Department of Electronics and Communication Engineering

School of Engineering, Tezpur University

Tezpur - 784028, Assam, India

2022-2023

Declaration

We hereby declare that the project work presented in this report entitled ***"Segmentation and Classification of skin lesion images using deep learning based techniques"***, submitted in partial fulfilment for the award of the degree of Bachelor of Technology in Electronics and Communication Engineering during the academic year 2022-2023, has been carried out by us and that it has not been submitted in part or whole to any institution for the award of any other degree or diploma.

Date:

Place:

(Devvjiit Bhuyan) (Md Noman)

(Gulshan Kumar) (Puralasetty Sumanth)

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
TEZPUR UNIVERSITY
Tezpur-784028, Assam, India

Prof. V. K. Nath **Phone: +91-3712-275264**
Professor **Email: vknath@tezu.ernet.in**

CERTIFICATE

This is to certify that the report entitled ***"Segmentation and Classification of skin lesion images using deep learning based techniques"*** submitted to the Department of Electronics and Communication Engineering, Tezpur University in partial fulfillment for the award of the degree of Bachelor of Technology in Electronics and Communication Engineering, is a record of project work carried out by ***Devvjiit Bhuyan (ECB19050), Md Noman (ECB19053) Gulshan Kumar (ECB19057), and Puralasetty Sumanth (ECB19060)*** under my supervision during the period from January 2023 to June 2023. All support received by them from various sources has been duly acknowledged. No part of this report has been submitted elsewhere for the award of any other degree or diploma.

Date: Prof. V. K. Nath
Place: (Supervisor)

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
TEZPUR UNIVERSITY
Tezpur-784028, Assam, India

Prof. S. Sharma
Head of the Department

Phone: 03712-275251
Fax: +91-3712-267005/6
Email: sss@tezu.ernet.in

CERTIFICATE

This is to certify that the report entitled ***"Segmentation and Classification of skin lesion images using deep learning based techniques"*** is a bonafide record of project work carried out by ***Devvjiit Bhuyan (ECB19050), Md Noman (ECB19053) Gulshan Kumar (ECB19057), and Puralasetty Sumanth (ECB19060)*** and submitted in partial fulfillment for the award of the degree of Bachelor of Technology in Electronics and Communication Engineering during the academic year 2022-2023. They have carried out their project work under the supervision of **Prof. V. K. Nath, Professor, Dept. of ECE, Tezpur University**.

This approval does not necessarily endorse or accept every statement made, every opinion expressed, or every conclusion drawn as recorded in the report. It only signifies the acceptance of this report for the purpose for which it is submitted.

Date:

Place:

Prof. S. Sharma

(HoD, ECE)

Certificate by the Examiner

This is to certify that the report entitled ***"Segmentation and Classification of skin lesion images using deep learning based techniques"*** submitted by ***Devvjiit Bhuyan (ECB19050), Md Noman (ECB19053) Gulshan Kumar (ECB19057), and Puralasetty Sumanth (ECB19060)*** in partial fulfillment of the requirements for the degree of Bachelor of Technology in Electronics and Communication Engineering has been examined by me and is found satisfactory for the award of the degree.

This approval does not necessarily endorse or accept every statement made, opinion expressed or conclusion drawn as recorded in the report. It only signifies the acceptance of this report for the purpose for which it is submitted.

Date: (Examiner)

Place:

Acknowledgements

We take this opportunity to acknowledge and express our gratitude to all those who supported and guided us during our project work. First and foremost, we are immensely grateful to our project supervisor **Prof. V. K. Nath**, for their invaluable guidance, expertise, and continuous support throughout the project. We also extend our sincere thanks to **Mr. Sunil Kumar**, Research Scholar, Department of ECE, Tezpur University for their valuable input, knowledge sharing, and encouragement. Their knowledge and advice have been really helpful in widening our comprehension of the subject area and raising the academic level of this research.

Devvjiit Bhuyan
Md Noman
Gulshan Kumar
Puralasetty Sumanth

Contents

List of Figures

List of Tables

List of Algorithms

Abstract

The project report aims to build a better supervised segmentation algorithm for dermopscopic images, with special emphasis on skin lesion images. For this task, the ISIC 2016 dataset is chosen, which comprises of high-resolution dermoscopy images of skin lesions. The objective is to extend the existing state-of-the-art technologies by implementing the Illumination-based Transformations along with a Unet architecture model with a DenseNet201 backbone. The report also shows the implementation of a Fully-unsupervised segmentation algorithm that is built using a simple ConvNet. The method uses multiple preprocessing stages, including hair removal and Normalized weighted contrast stretching. Oversegmented images are generated using a modified version of the CNN, which are then post-processed in multiple stages, starting with a marginalization algorithm that will remove smaller irrelevant regions from the image, then a Region merging algorithm is run through the filtered image. Finally, a binary mask is returned where white denotes the lesion region and black denotes the background. The modules used in the project include the usage of Illumination-based Transformations, a Unet architecture model with a DenseNet201 backbone, and a Fully-unsupervised segmentation algorithm. The project also utilizes popular performance enhancement techniques such as Augmentation, Image Quality enhancement, Hair removal, etc. The supervised method has shown steady improvement for the IoU and Dice score metrics against the best existing models. The unsupervised method also shows some promising results. Additionally, the work is followed by a classification algorithm based on SVM and Attention-mechanism to accurately diagnose skin cancer.

1. Chapter: Introduction

The field of medical imaging has experienced remarkable advancements in recent years, with a growing emphasis on leveraging machine learning techniques for accurate and efficient image analysis. One of the crucial tasks in medical image analysis is segmentation, which involves identifying and delineating specific regions of interest within an image. Accurate segmentation plays a pivotal role in various applications, including disease diagnosis, treatment planning, and monitoring treatment efficacy. The ISIC 2016 dataset serves as the focus of this comprehensive study on supervised and unsupervised segmentation. It comprises of high-resolution dermoscopy images of skin lesions. Dermoscopy is a way to take pictures of the skin without hurting it. It helps doctors figure out if someone has skin cancer or other problems with their skin. The ISIC 2016 dataset consists of 1279 dermoscopic images, which have been manually annotated by dermatologists to delineate the boundaries of skin lesions. The availability of ground truth annotations makes this dataset suitable for both supervised and unsupervised segmentation approaches. To enhance the segmentation performance, several preprocessing techniques are applied to the dataset, including dullRazor [18], and Illumination based Transformations [2]. Data augmentation techniques are used to make the training data more diverse. This helps to increase the variety of information that the data contains. Methods such as rotation, scaling, and flipping are employed to create additional training samples. Augmentation aids in improving the robustness and generalization capability of the segmentation models. Illumination-based transformations address variations in lighting conditions across the dataset. These transformations aim to normalize the image intensities, ensuring consistent illumination characteristics and minimizing the impact of lighting variations on the segmentation performance. The supervised learning architecture, U-Net, is utilized for both supervised segmentation of the ISIC 2016 dataset.

The U-Net design has become very popular for analyzing medical images because it can understand the overall context while still preserving the small details. It has two main parts: the encoder and the decoder. The encoder gathers important information in layers, and the decoder uses this information to create a detailed map for segmenting the image. In this project, the use of the DenseNet201 [9] architecture as a backbone for the U-Net model is investigated. DenseNet is renowned for its dense connectivity patterns, which facilitate feature reuse and enhance the flow of information across different layers. By incorporating DenseNet as a backbone, the goal is to exploit its powerful feature extraction capabilities and further improve the segmentation performance. Additionally, for latter datasets such as the ISIC 2018 dataset, there are no predefined ground-truth images for segmentation, hence we have experimented with another Fully-Unsupervised method which doesn't require any ground truth masks. It has been tested on the ISIC 2016 dataset to compare it's efficacy. Cancer is the leading cause of mortality with 9.6 million fatalities in 2018.

Cancer is mostly brought on by three factors: lifestyle, environment, and genetic issues. Cancer that is primarily internal and cannot be seen with the unaided eye

or on the skin may be caused by a somatic mutation. The most prevalent form of cancer that can be brought on by radiation, UV light, or pathogens is skin cancer. Detecting skin cancer is difficult because it can appear in various ways. Many abnormal changes in tissues and cells can be signs of skin cancer, but it's hard to tell if the changes are cancer or not. If skin cancer is not caught early, it can spread to other parts of the body and become very hard to treat. These cancers can spread to other organs and harm healthy tissues if detected too late. Skin cancer is difficult to detect with the naked eye since it starts off extremely little, like a mole. Early skin cancer diagnosis has been achieved by dermoscopy. It is a non-invasive diagnostic method for the assessment of non-pigmented and pigmented skin lesions that cannot be seen with the unaided eye. On the basis of their strong representational power, convolutional neural networks (CNNs) have greatly improved the performance of vision tasks. Recent studies have focused on three critical network properties: depth, width, and cardinality, with the goal of improving CNN performance. Yet, if there were computer software that could instantly identify skin cancer from a digital photograph taken using any digital image-capturing device The victim can do the test at any time, including at home, using a system that places little emphasis on the area of interest. Skin conditions can be broadly divided into two categories: benign and malignant. The early stage of cancer is known as benign, but the advanced stage is known as malignant. The identification of skin cancer poses significant challenges due to its resemblance to other skin conditions. To tackle this problem, classifying skin cancer from dermoscopic images is treated as an image classification task. Traditional image classification techniques rely on extracting distinct features from images, such as texture, color, or shape, which are then used to train the classifier. However, in the case of skin cancer, extracting and categorizing images based on these features is difficult. As a result, researchers have turned their attention to deep convolutional neural networks (CNNs) for feature extraction [10].

Recent research has shown that CNNs (Convolutional Neural Networks) are very good at automatically finding important details in images. They can identify both macro features like the meaning and texture of an image, as well as smaller details like edges and shapes. Many scientists have been studying different ways to use CNNs in skin cancer research. They have been working on tasks like figuring out the boundaries of skin cancer areas, finding and recognizing cancer cells, and categorizing different types of skin cancer. They use techniques from fields like classification, image processing, machine learning, computer vision, and deep learning to tackle these challenges. The CBAM module, as described by Sanghyun et. Al.[22], consists of two sub-modules: channel and spatial. This module is employed sequentially to adaptively refine the intermediate feature map within each convolutional block of deep networks.

The work done in this project is threefold:

1. Supervised Segmentation for dermoscopic images based on the U-Net architecture with a DenseNet201 backbone alongwith Illumination-based Transformations

2. Fully-Unsupervised Segmentation based on simple CNN and a Region Merging Algorithm, and

3. Skin lesion classification for accurate diagnosis of cancer using Transfer learning and Attention based methods.

2. Chapter: Related Work

2.1 Unsupervised learning

DullRazor, a pre-processing algorithm introduced by Gallagher et. Al.[18] in 2007, has emerged as a valuable tool in the field of dermoscopic image analysis. The primary objective of DullRazor is to eliminate the presence of hair in dermoscopic images, which can confound segmentation algorithms and compromise the diagnostic process. Thick, dark hairs in dermoscopic images often possess similar characteristics to actual skin lesions, making them challenging to distinguish. DullRazor employs advanced algorithms and techniques to identify and remove these hair artifacts, thereby reducing interference and improving the accuracy of subsequent segmentation processes.

Kanezaki et al.[21] proposed an algorithm for skin lesion segmentation based on images of humans, animals, and household items, which has gained significant attention in recent years. The algorithm utilizes a deep convolutional neural network (CNN) to extract relevant features from the images. These extracted features are then subjected to hierarchical clustering, a technique that groups similar features together to form distinct clusters. By leveraging the power of deep learning and clustering algorithms, Kanezaki's algorithm can effectively segment dermoscopic images into regions corresponding to different skin lesions. To adapt the algorithm specifically for dermoscopic images, the researchers incorporated the CNN into the PASCAL VOC dataset.

In another work by Ali. et. Al.[1], the application of Kanezaki's method to the ISIC dataset was explored. This work demonstrated the effectiveness of Kanezaki's algorithm in both supervised and unsupervised scenarios, further solidifying its potential as a versatile tool for skin lesion segmentation. By leveraging the vast amount of labeled data available in the ISIC dataset, researchers were able to train the algorithm to accurately segment skin lesions. Additionally, the unsupervised approach allowed the algorithm to generalize to new, unseen data, making it applicable to a wide range of dermoscopic images.

Khan et. Al.[11] proposed another notable research work in the field of skin lesion segmentation. This work focused on the utilization of the Region Merging algorithm, which integrates the concept of High-Dimensional Feature Vectors (HDFV). The Region Merging algorithm aims to refine the initial segmentation results by iteratively merging regions with similar features. By incorporating HDFV, which captures both local and global features, the algorithm achieves improved accuracy in segmenting skin lesions. The integration of deep learning-based representation clustering further enhances the algorithm's performance, providing more robust segmentation results.

In the quest for effective texture feature descriptors, Verma et. Al.[20] proposed the CSLBP based Gray-Level Co-occurrence Matrix (GLCM). The GLCM is a way to study how pixels in an image are connected to each other. It looks at the patterns of texture in the image and uses that to find important details for separating different parts of the skin. The GLCM-based method has been shown to be really good at

capturing the texture of skin lesions. It can measure things like contrast, homogeneity, and entropy, which helps in understanding the characteristics of dermoscopic images. This makes it a valuable tool for analyzing skin problems.

2.2 Supervised learning

The UNet architecture [16] derives its name from its U-shaped structure, consisting of two key components: The encoder, often referred to as the contraction path, and the decoder, commonly known as the expansion path, constitute integral components of the architecture. The encoder is responsible for capturing the context within the image, extracting high-level features that represent the spatial information. On the other hand, the decoder utilizes these extracted features to reconstruct the spatial information, generating a segmented output that accurately identifies the regions of interest.

The essential aspect of the UNet architecture is its U-shaped design, which plays a crucial role in efficiently capturing contextual information within the image and closing the semantic gap between the feature maps of the encoder and decoder. By doing so, the UNet architecture overcomes the limitations of traditional fully convolutional networks, which often struggle with preserving fine-grained details during the encoding and decoding process.

Building upon the success of the UNet architecture, researchers have introduced a modified version known as UNet++ [29]. UNet++ incorporates the concept of Dense blocks from DenseNet [9], a state-of-the-art architecture for image classification, to further enhance its performance in biomedical image segmentation tasks. Dense blocks promote feature reuse and improve gradient flow, leading to more accurate and precise segmentation results. The integration of Dense blocks in UNet++ brings several benefits. It enables the network to efficiently leverage the collective knowledge from earlier layers, enhancing the segmentation accuracy. Additionally, Dense blocks contribute to bridging the semantic gap between the encoder and decoder feature maps, allowing for seamless information flow throughout the architecture. To boost the performance of the UNet architecture, researchers have explored the incorporation of backbones. Backbones are pre-existing deep neural networks that serve as feature extractors, providing a strong foundation for subsequent layers. By integrating backbones into the UNet architecture, multiple advantages can be obtained. Firstly, backbones help improve the overall segmentation accuracy of the UNet architecture by leveraging the pre-trained weights and learned representations of the backbone network. This transfer learning approach enables the UNet architecture to benefit from the knowledge gained during the training of the backbone network, resulting in more robust and accurate segmentation. Secondly, backbones assist in bridging the semantic gap between the encoder and decoder feature maps. The use of a well-designed backbone network ensures that the high-level features extracted by the encoder are efficiently propagated to the decoder, facilitating the reconstruction of spatial information with improved accuracy. Lastly, backbones contribute to

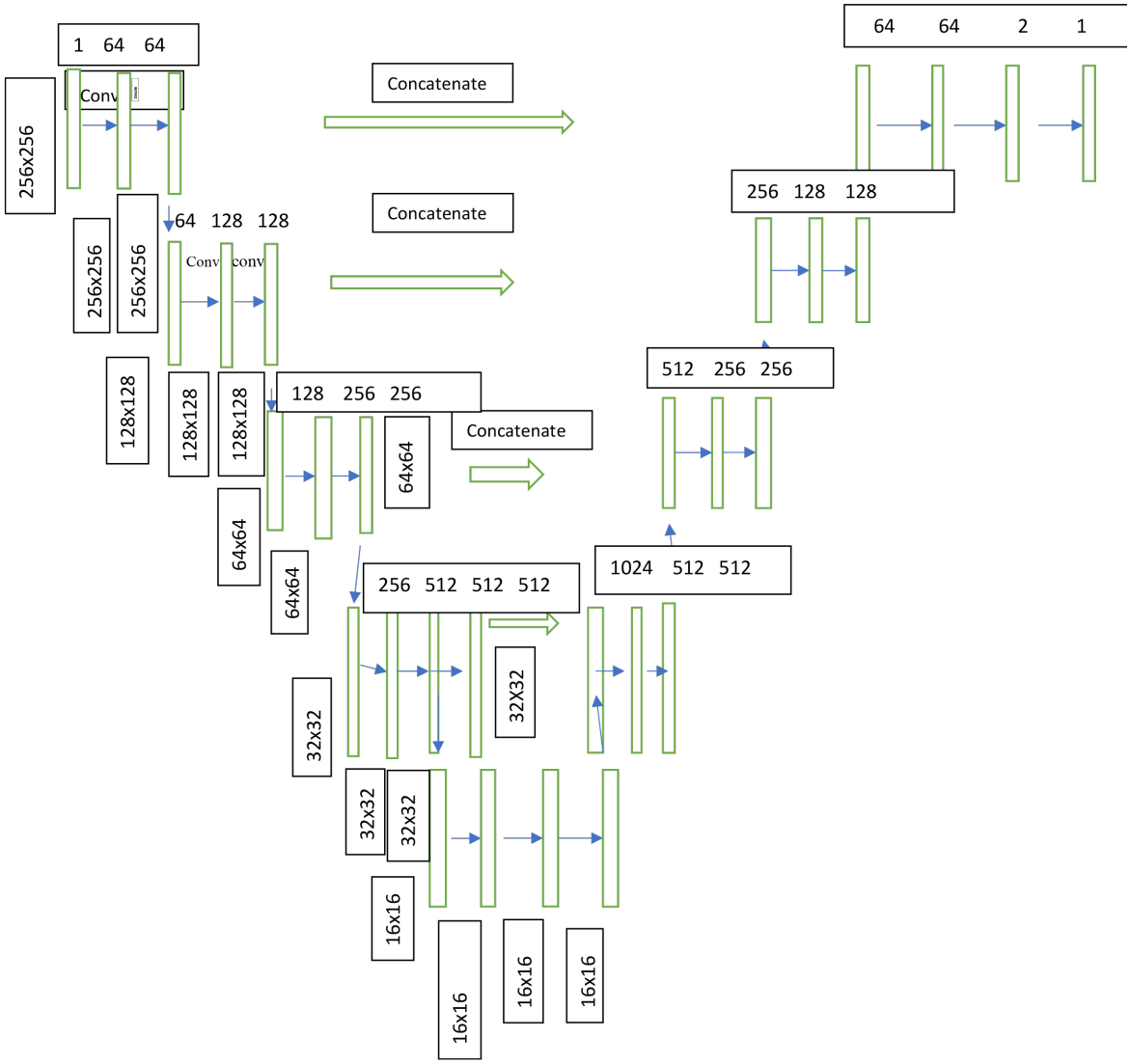

Figure 1: **U-Net architecture**

improving the gradient flow within the UNet architecture. With convolution layers on skip pathways and dense skip connections, the incorporation of backbones enables seamless information propagation, reducing the likelihood of information loss or degradation during the segmentation process.

Data augmentation techniques play a pivotal role in training robust segmentation models. By augmenting the training dataset, we can increase its diversity, which helps the model generalize better to unseen data. Perez et. Al.[14] demonstrated the positive impact of augmentation on segmentation models. The mentioned paper focuses on the ISIC 2018 Skin Lesion Segmentation challenge dataset. The authors showcase how applying techniques like rotation, flipping, and scaling to the training images

significantly improves the segmentation model's performance. Data augmentation augments the training set, effectively expanding the available data for the model to learn from. This leads to better generalization, improved accuracy, and robustness in segmenting skin lesions.

In the pursuit of enhancing segmentation model performance, researchers have explored the role of shading attenuation. Zhang et. Al.[28] proposes an intriguing approach to improve segmentation models. The paper suggests that by attenuating the shading in an image, the segmentation model can better distinguish between different regions. The attenuation of shading effectively reduces the impact of lighting conditions on the segmentation task. By mitigating the variations caused by illumination, the model can focus more on the intrinsic properties of the objects being segmented. This technique enhances the model's ability to differentiate between regions with similar color intensities but distinct structures, leading to more accurate and precise segmentation results.

Abhishek et. Al.[2] proposed another paper that shows another avenue for improving segmentation model performance. The paper suggests that illumination-based transformations, such as brightness and contrast adjustments, can significantly enhance the intersection over union (IoU) and Dice scores of the segmentation models. By manipulating the illumination conditions in the training images, the segmentation model becomes more resilient to variations in lighting during inference. The brightness and contrast adjustments modify the image appearance, thereby enriching the diversity of the training data. As a result, the model learns to adapt to different lighting conditions, leading to improved generalization and robustness in segmenting medical images. In conclusion, the UNet architecture has revolutionized biomedical image segmentation by providing a robust and efficient solution.

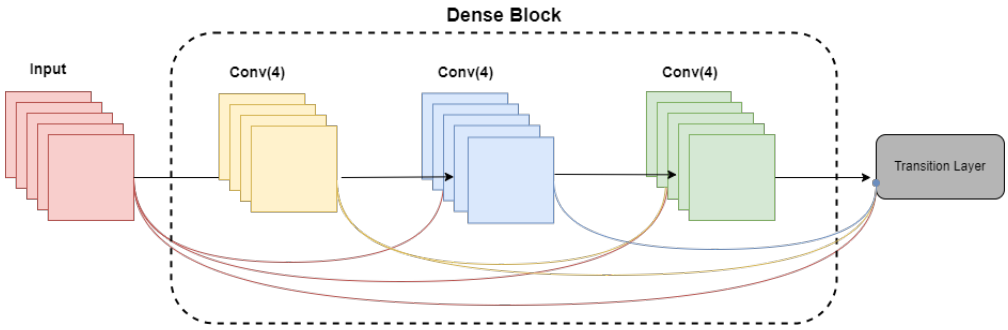

Figure 2: **DenseNet architecture**

The integration of backbones, such as DenseNet, enhances the UNet architecture's performance by leveraging pre-trained features. Additionally, data augmentation techniques, including rotation, flipping, and scaling, augment the training set and improve generalization. Shading attenuation and illumination-based transformations further contribute to better segmentation results.

2.3 Classification

Classifying data into different classes or groups based on specific traits or properties is a crucial activity in machine learning and data analysis. It is a method of supervised learning where a model is trained on a labelled dataset where each occurrence of the data is linked to a predetermined class label.

Creating a predictive model that can generalise from the labelled training data to correctly categorise new, unforeseen cases into the appropriate categories is the aim of classification. During the training phase, the model discovers patterns or correlations between the input features and the corresponding class labels. After being trained, the model can make predictions on fresh, unlabeled data by classifying it.

Many domains and issue kinds can be classified, such as:

- Image classification is the process of categorising photographs, such as by identifying items or spotting patterns in them.
- Medical diagnosis is the process of determining a patient's condition or disease using imaging or medical record data.
- Text classification is the process of grouping text content into preset categories, such as subject classification or sentiment analysis.
- Differentiating between authentic and spam emails or communications is known as spam detection.
- Identifying the likelihood of default or the creditworthiness of loan applicants is called credit risk assessment.
- Fraud detection is the process of identifying questionable behaviour or transactions using previous data and patterns.

Typical classification methods include:

- SVMs, or support vector machines, create hyperplanes to group data points into several classes.
- Decision Trees: Models that resemble trees that divide the data into categories depending on attributes to create forecasts.
- Random Forests: Decision tree ensembles that mix several models for greater accuracy.
- Using a logistic function, logistic regression calculates the likelihood that a given instance belongs to a given class.
- Deep learning models called neural networks are made up of interconnected layers of synthetic neurons.

According to Yu. et. Al.[25], performing classification directly using CNN features would be quite difficult for images with sharp differences in viewpoint and resolution. Solutions that are frequently employed include rescaling and data augmentation (crop, flip, or rotate) method. Unfortunately, some data modifications may have a negative impact on performance. For instance, randomly cropped photos might only capture a background region without the item or a non-interesting portion of the object in the original image, exposing the classifier to unhelpful representations. The performance boost is hardly noticeable. The situation grows worse when CNN is used for medical purposes.

Younis et. Al.[24] developed a multi-class strategy for classifying dermoscopic images of skin cancer into one of the seven malignancies. The Harvard HAM1000 skin cancer dataset, which contains 10015 dermatoscopic pictures, is used to fine-tune the MobileNet convolutional neural network, which was previously trained on 1.3 million photos. To better understand skin cancer, they also looked at the information to see how skin lesions related to various criteria.

A straightforward deep learning technique called transfer learning makes use of the initialization weights of pretrained networks that have been trained on different datasets. ResNet50, DenseNet, and MobileNet were used as separate transfer learning models that were refined and combined to improve classification accuracy and resilience, however the results were not up to par.

Support Vector Machines (SVMs) have gained extensive usage alongside Convolutional Neural Networks (CNNs) in the field of computer vision, serving various purposes. Numerous existing studies have delved into this combination, and the following literature highlights some notable works in this domain [4].

- Seeja R.D. and Suresh A. [17] present a methodology that combines deep learning and SVM for skin lesion segmentation and classification. The proposed approach involves utilizing a deep learning model to accurately segment skin lesions. Subsequently, feature extraction is performed on the segmented Region of Interest (ROI), enabling the differentiation of malignant melanoma from benign lesions. This feature extraction process effectively reduces the size of the dataset, focusing on a concise and relevant subset that facilitates accurate classification with a high level of precision. While previous studies have investigated color, lesion boundary, and texture features to enhance system performance, the proposed method incorporates color, texture, and shape features. By leveraging these diverse features, the system aims to improve the overall accuracy and effectiveness of skin lesion segmentation and classification.

- Another approach from Polat et. Al.[4] combines deep learning models for both feature extraction and classification tasks. The proposed method outlines an effective deep network architecture that combines different structured deep models. Through experimental studies, the proposed approach has demonstrated exceptional performance in the classification of melanoma.The paper discusses

a method that uses advanced image improvement techniques and existing neural networks to identify different types of skin lesions. Instead of creating a brand new neural network, the method uses already trained networks and their weights to classify the lesions. This summary captures the main ideas from the original text in a slightly different way.

Sanghyun et. Al.[22] uses the channel attention model and spatial attention model for Convolutional Block Attention Module (CBAM). The channel attention model is a component commonly used in computer vision tasks, especially in the field of deep learning and convolutional neural networks (CNNs). It is designed to improve the representation power of CNNs by explicitly modeling the interdependencies between channels. In a CNN, each layer consists of multiple channels, and each channel represents a specific feature or pattern that the network has learned to detect. The channel attention model aims to dynamically adjust the importance of each channel based on its contribution to the task at hand. The channel attention mechanism typically involves two key steps: channel-wise feature aggregation and channel-wise feature recalibration. In the feature aggregation step, information from all the channels is combined to form a global context vector. This is often achieved by applying global pooling operations, such as average pooling or max pooling, across spatial dimensions. Once the global context vector is obtained, the feature recalibration step takes place. This step uses various techniques to transform the global context vector into a channel attention map. This attention map assigns a weight or importance value to each channel, indicating how much it should contribute to the final representation. Next, the channel attention map is employed to the initial feature map through element-wise multiplication, thereby enhancing the feature map. This recalibration process amplifies important channels while suppressing less relevant ones, effectively highlighting the most discriminative features for the task. By adding channel attention models to CNN structures, significant progress has been made in various computer vision tasks, including image classification, object detection, and image segmentation. This has allowed researchers to achieve the best performance currently available. These models help the network focus on the most informative channels, leading to improved accuracy and generalization capabilities.

The spatial attention model, also known as the spatial attention mechanism, is another component commonly used in computer vision tasks, particularly in deep learning and convolutional neural networks (CNNs)[15]. It is designed to selectively emphasize or suppress spatial regions within an image, allowing the network to focus on the most relevant parts for the task at hand. In CNNs, spatial attention models aim to capture the spatial relationships and dependencies between different regions of an image. This is particularly useful in scenarios where certain regions contain more important or discriminative information than others. The spatial attention mechanism typically involves the following steps:

1. Computation of spatial attention weights: The first step is to compute spatial

attention weights that indicate the importance or relevance of each spatial location. This is often done by applying certain operations, such as convolutions or pooling, to the feature maps extracted from the CNN layers.

2. Normalization of attention weights: To ensure that the attention weights are normalized and sum up to 1, a normalization step is performed. This step ensures that the attention weights can be interpreted as a probability distribution.

3. Modulation of feature maps: The attention weights are then used to modulate or scale the feature maps obtained from the CNN layers. The modulation is typically performed by element-wise multiplication, where each feature map value is multiplied by its corresponding attention weight.

   The rationale behind this approach lies in the notion that the convolutional layers possess the ability to extract general, low-level features that have broad applicability across different images. These features encompass fundamental elements such as edges, patterns, and gradients. In contrast, the subsequent layers in the network are responsible for identifying specific features within an image, such as eyes or wheels.

   To summarize, the general outline for transfer learning in object recognition, as proposed by Koehrsen et. Al.[12], can be outlined as follows:

   (a) Utilize a pre-trained model on a large dataset.

   (b) Freeze the initial convolutional layers of the network.

   (c) Focus on training the last few layers responsible for predictions.

   (d) Leverage the general, low-level features extracted by the convolutional layers.

   (e) Employ the subsequent layers to identify specific features within images.

3. Chapter: Methodology

3.1 Supervised Segmentation

This section explains the work done on Supervised Segmentation, where the ground truth images were available. The Supervised Segmentation approach is visualized in fig. (4). It consists of two stages: preprocessing, and learning. The preprocessing tasks include

- Hair Removal and Image Quality improvement using DullRazor
- Affine Augmentation
- Illumination-based Transformations

The preprocessed images being fed into a supervised learning model based on the U-Net architecture. Each of the steps is covered elaborately in the following sections.

3.1.1 Datasets

Figure 3: **Sample images from the ISIC 2016 dataset**

We have used the ISIC 2016 and 2017 segmentation datasets published by the International Skin Imaging Collaboration (ISIC). It was part of a challenge at the International Symposium on Biomedical Imaging (ISBI) 2016 and was designed to evaluate algorithms for skin lesion segmentation. The 2016 dataset consists of 900 training images and 379 testing images, with corresponding segmentation masks for each image. For the segmentation objective, we consider all images regardless of class distribution, while maintaining the separate train

Table 1: **Class Distribution for the ISIC 2016 dataset**

Class	*Test*	*Train*	*Masks – available*
benign	304	727	Yes
malignant	75	173	Yes
Total	379	900	

Table 2: **Class Distribution for the ISIC 2017 dataset**

Class	*Test*	*Train*	*Masks – available*
benign	393	1372	Yes
malignant	117	374	Yes
seborrheic keratosis	90	254	Yes
Total	600	2000	

and test images. The ISIC 2017 dataset contains 2600 images divided into 3 classes: benign, malignant, and seborrheic keratosis. The class distribution is shown in Table (2).
For the classification task, the Test and Train data are divided into classes, with a huge class imbalance, as shown in Tables (1) and (2).

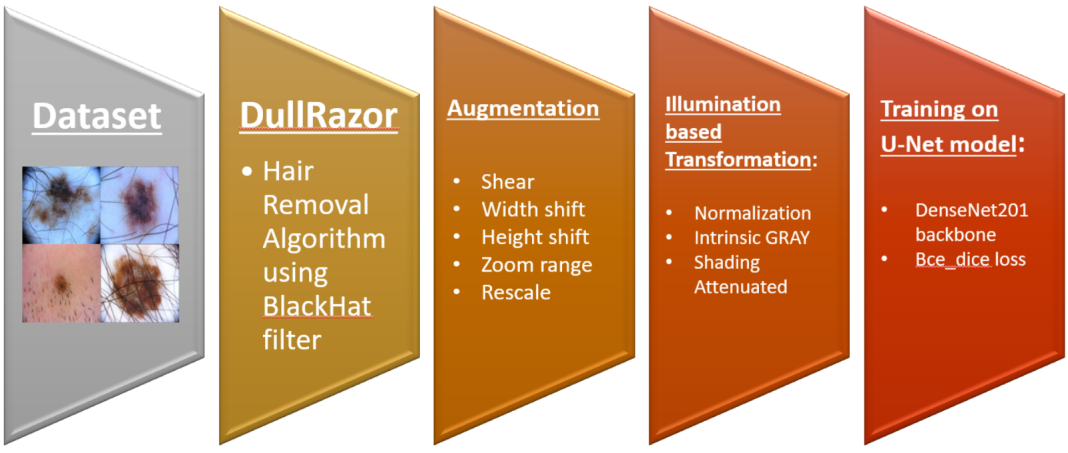

Figure 4: **Flowchart Showing the Supervised Segmentation algorithm**

Figure 5: **Some images having Hair occlusions in the ISIC 2016 dataset**

3.1.2 DullRazor

The DullRazor [18] algorithm is a pre-processing program used to remove hair from images. It applies a series of morphological operations to the image to generate a mask that contains the hairs. The algorithm is specifically designed for dermoscopic images and is used as a pre-processing step to enable better lesion segmentation. The image is passed through a black-hat filter, designed to highlight the darker structures (hair) in the image, this output is blurred and thresholded and inpainting is done based on this. The resultant image is free from all hair or similar structures. Fig (5) shows various images where the lesion can be seen obstructed by hair, this hair can perturb the segmentation algorithm's feature extraction process causing the segmentation algorithm to give inaccurate results.

Table 3: **Augmentation settings**

zoom_range	0.2
shear_range	0.2
height_shift_range	0.2
width_shift_range	0.2
rotation_range	90
horizontal_flip	True
rescale	1./255

3.1.3 Augmentation

After processing every image through the DullRazor algorithm, we Augment the training set of images owing to the small size of the dataset. The augmentations that we have used are shown in table (3). These are performed explicitly on the dataset to reduce overhead while training, such that the number of training images is 1895 for the 2016 dataset (5*number of test images), and 3000 for the

2017 dataset. Masks for augmented images were generated iteratively by using a model trained on the smaller original dataset.

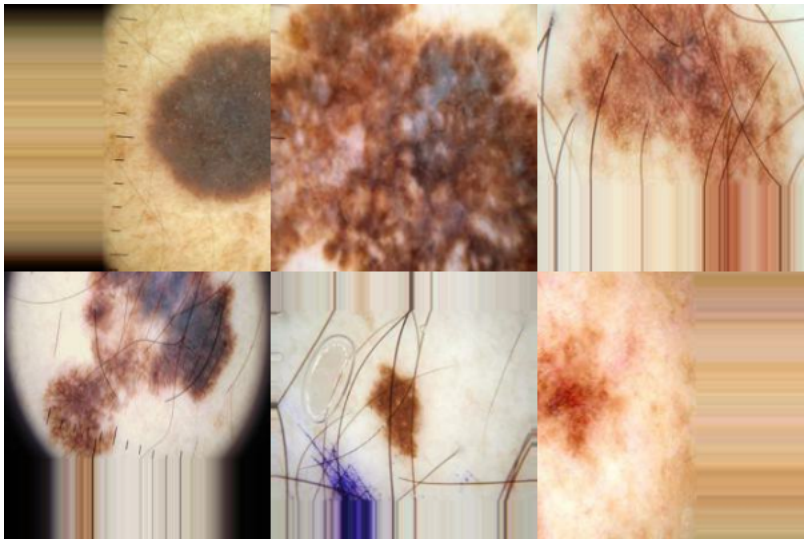

Figure 6: **Images after various augmentations**

3.1.4 Illumination based Transformation

The importance of lighting and other similar physics-based features in skin lesion images is frequently ignored by deep learning-based techniques. Abhishek et. Al.[2] proposed a color-theory-based approach to improve the visibility and contrast between the lesion and the surrounding skin, which has been proven to improve the overall segmentation performance of deep learning models. This approach uses a multiple methods, starting with intensity normalization and histogram equalization, creating an Illumination-invariant grayscale estimate using Singular Value Decomposition (SVD), and a Shading-Attenuated [28] representation for the skin lesion. These image-maps, alongwith the original RGB channels are concatenated and sent into the segmentation algorithm. The distribution of channels can be seen in Fig. (7).

3.1.5 The U-Net architecture

The U-Net architecture is a special kind of neural network that was made specifically for separating different parts of biomedical images. It has two main parts: the contracting path, which acts like an "encoder" and the expansive path, which acts like a "decoder". The encoder is used to capture the context in the image, while the decoder is used to recover the spatial information lost during the encoding process. The main idea behind U-Net is to supplement a usual

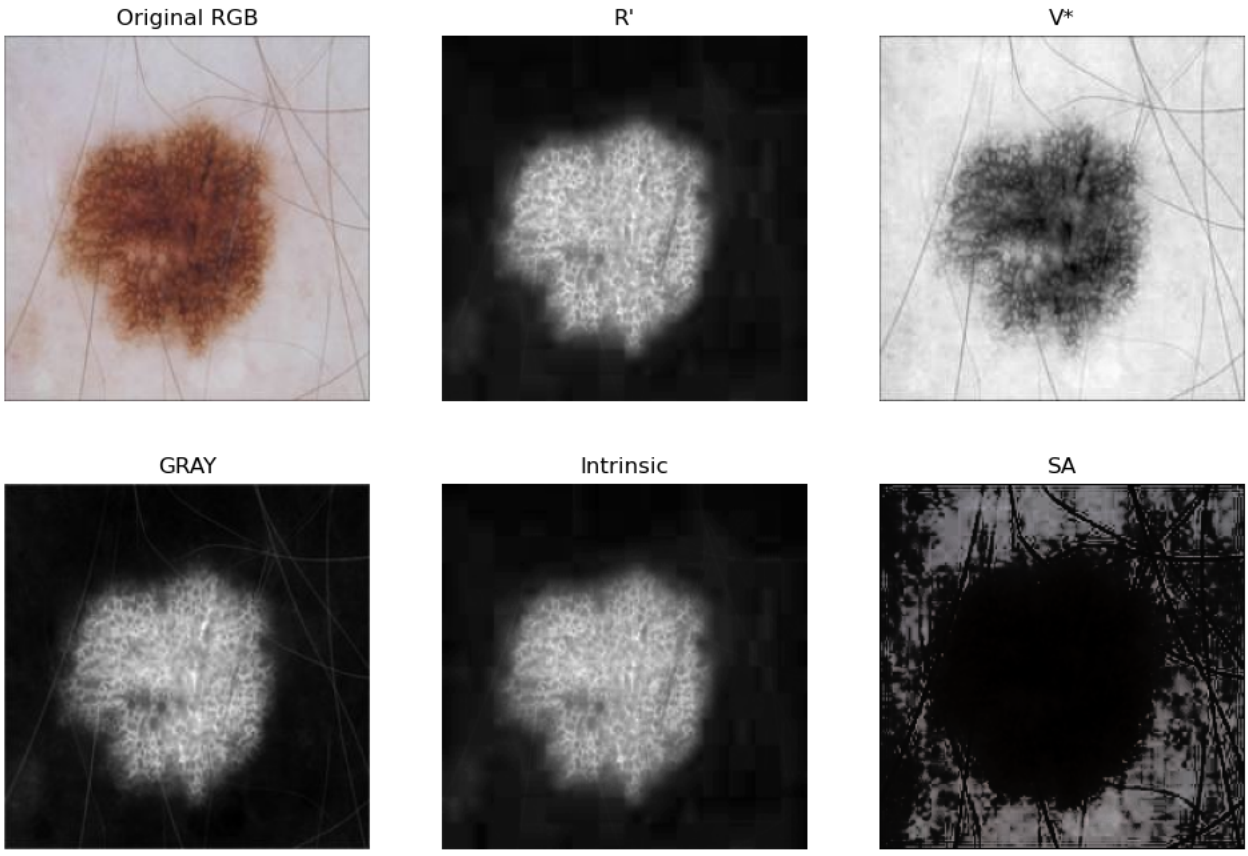

Figure 7: **Images after Illumination-based Transform. LT: Original, CT: Red-Normalized, RT: Max-Normalized, LB: Intrinsic Grayscale, CB: Illumination-Invariant, RB: Shading-Attenuated**

contracting network by successive layers, where upsampling operators replace pooling operations. This allows the network to increase the resolution of the output and learn to assemble a more precise output based on this information. U-Net is able to precisely localize and distinguish borders and performs classification on every pixel so that the input and output have the same resolution. The benefits of U-Net include its ability to work with fewer training images and yield more precise segmentation, as well as its computational efficiency and ability to learn segmentation in an end-to-end setting.

3.1.6 The DenseNet architecture

DenseNet201 is a very deep neural network that has 201 layers. It's an extension of the DenseNet architecture, which uses special connections between layers called "dense" blocks. In these blocks, layers with the same size of features

are connected to each other directly. Each layer also gets information from all the previous layers and shares its own information with the subsequent layers, keeping the flow of information going forward. The advantages of DenseNet201 are that it can achieve high accuracy while using fewer parameters compared to other deep neural networks like ResNet and Pre-Activation ResNet. It also works well even when there isn't much training data available because it uses features at different levels of complexity.

We have used the U-Net model with a DenseNet201 backbone. The list of hyperparameters and model specifics are listed in Table (6) These were found to be the optimal values after several experiments.

3.2 Unsupervised Segmentation

This module explains the work done on Unsupervised Segmentation, the case where we assume that the Ground Truth Images are unavailable (which is the case with the 2018 dataset). Here also, we have chosen to work on the same ISIC 2016 dataset, such that the results can be objectively compared with the existing Ground Truth images. This project extends on the work done by Kanezaki et. Al.[21], who proposed an Unsupervised Segmentation algorithm based on differentiable feature clustering. Although it was originally intended and tested on the PASCAL VOC dataset, which contains images of people, animals, and household items. This version is built to work with dermoscopic images. The sequential working of this algorithm is explained as follows

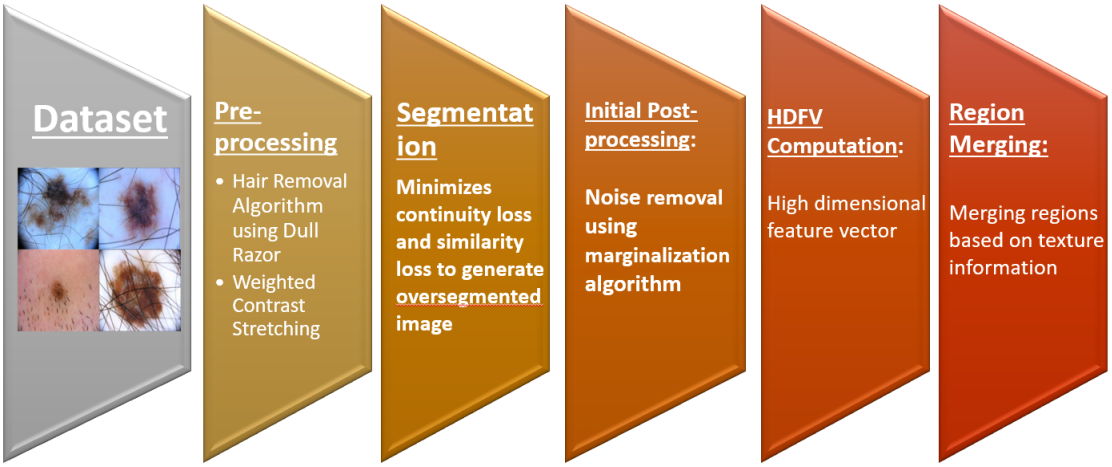

Figure 8: **Unsupervised Segmentation flowchart**

3.2.1 Preprocessing

As with the supervised use-case, we firstly remove all hair and similar occlusions from all the images using the DullRazor algorithm [18]. The $R-Normalized$ [2] image-map is computed and **weighted contrast stretching** is done to improve the visibility of the lesion to the Convolutional Network. No augmentation is done in this case as the entire concept is around Unsupervised Segmentation.

3.2.2 Segmentation

The algorithm proposed by Kanezaki et. Al.[21] is built upon two main optimization criteria: similarity loss and continuity loss. Pixels with similar color schema must be grouped together, and pixels adjacent to each other must be

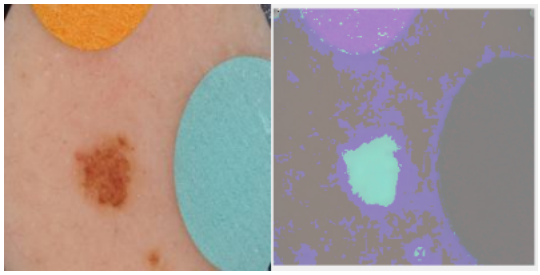

Figure 9: **L: Image before, and R: after weighted contrast stretching**

grouped together. The model was a ConvNet with 3 convolutional layers. Cross entropy loss is considered as the similarity loss and L1 loss along the x and y directions is considered the continuity loss. Overall, a weighted sum of both losses is optimized over a number of epochs. The weights are tuned to our use case.

$$loss = 4 \times crossentropyloss + 0.25 \times (lhpy + lhpz) \tag{1}$$

where *loss* is the total loss being optimized in every iteration, *crossentropyloss* is the similarity loss (set to a higher weightage), and *lhpy* and *lhpz* are the continuity losses in the y and z directions respectively (or x/y, based on the cartesian coordinate system).

The above method returns an over-segmented image with a huge number of unique regions. Which is then post-processed by the Region-Merging algorithm.

3.2.3 Initial Post-processing

All over-segmented images are passed through a marginalization algorithm, where any region smaller than a certain threshold is merged into its surrounding regions. The marginalization algorithm in itself has a number of smaller functions which tackle different kinds of artefacts. This is explained in more detail in the algorithm below.

3.2.4 HDFV computation

For the Region Merging Algorithm, we consider the use of an HDFV(High Dimensional Feature Vector), which is taken up elaborately in this section. The HDFV is a 1-dimensional vector of length 6912. The RGB histograms make up the first 768 values of the HDFV, while the remaining blocks contain the VECT, which is defined as follows GLCM(Gray Level Co-occurrence Matrix) [20] is computed for the CSLBP(Centre-Symmetric Local Binary Pattern) map of individual R, G, B channels as well as combinations of channels. Due to the

Algorithm 1 Marginalization Algorithm

Input: *img* is the raw over-segmented image returned by the CNN, *thresh* is the threshold ratio of the area below which any speck (tiny region) will be erased

Output: *img* is the filtered mask after initial marginalization, which will be fed to the Region Merging Algorithm as $unfiltered$

```
def marginalize(img, thresh = 0.001):
    mask = array of zeros with shape (224, 224)
    masks = [mask]
    img = remove_noise(img)
    img = remove_edge_artefacts(img)
    threshold = thresh multiplied by 224 multiplied by 224
    flag = 1
    for i in range of img.shape[0]:
        for j in range of img.shape[1]:
            start_pos = [i, j]
            buf = img
            if maximum value of [m[i, j] for m in masks] is True:
                continue
            else:
                mask, ar = area(img, start_pos)
                if ar is less than or equal to threshold:
                    c = largest_boundary(img, mask)
                    if c is not None:
                        img = fill_color(img, start_pos, c, mask)
                    else:
                        raise Exception('Inaccurate boundary at pos: ', start_pos)
                masks.append(mask)
                num_regions, final_masks = check_num_regions(img)
            if num_regions is less than 3:
                img = buf
                flag = 0
                break
        if flag is not 0:
            break

    return img
```

nature of dermoscopic images (which may come in varying color schemes), we focus more on the texture part, where GLCM and CSLBP come to the forefront as feature descriptors for textures. Another observation was that combinations of individual channels sometimes prove to have better contrast and show much

Table 4: **Composition of our HDFV**

Attribute	*Length*($flattened$)
Blue-channel Histogram	256
Green-channel Histogram	256
Red-channel Histogram	256
GLCM 0^{o} of CSLBP (Blue-channel)	256
GLCM 45^{o} of CSLBP (Blue-channel)	256
GLCM 90^{o} of CSLBP (Blue-channel)	256
GLCM 135^{o} of CSLBP (Blue-channel)	256
$Total - length - of - GLCM - in - 4 - directions$	1024
GLCM in all 4 degrees of CSLBP (Green-channel)	1024
GLCM in all 4 degrees of CSLBP (Red-channel)	1024
GLCM in all 4 degrees of CSLBP (Red+Blue channel)	1024
GLCM in all 4 degrees of CSLBP (Red+Green channel)	1024
GLCM in all 4 degrees of CSLBP (Blue+Green channel)	1024
$Total - Length - of - vector$	6912

better disparity between the lesion and the background, hence we have computed the feature maps of individual channels, as well as binary combinations of channels.

3.2.5 Region Merging Algorithm

The inspiration for this is derived from the work done by Khan et. Al.[11], where they proposed the usage of RAG (Region-Adjacency Graph) and an HDFV (High-Dimensional Feature vector) based Region merging algorithm for finer segmentation. Our HDFV, shown in Table. (4), differs from theirs in the way that we focus more on the texture information rather than the spatial or color information. In our algorithm, depicted in Alg. (2), we iterate through the different regions in an over-segmented image. To select a region, we use the flood-fill algorithm. For every region, we compute the HDFV, which contains information about the color schema, and texture information for the corresponding region. Based on these HDFVs, vector distance is computed among the adjacent regions, the regions with the smallest vector distance are merged

together. A new HDFV is computed for the newly merged region. This goes on until there are only two large regions in the image. Following that, the filtered segmented output is binarized into a black-and-white image with white denoting the lesion mask and black denoting the background.

Algorithm 2 Region Merging Algorithm

Input: $unfiltered$ is the over-segmented image returned by the CNN which has been processed by the marginalization algorithm, img is the original RGB image of the skin lesion.

Output: $filtered$ is the filtered mask after region merging

```
def rm(unfiltered, img):
    num_regions, masks = check_num_regions(unfiltered)
    adj_hdfvs = [hdfv(img, mask2) for mask2 in masks]

    while len(masks) > 2:
        mask_hdfv = adj_hdfvs[0]
        distances = [manhattan_distance(mask_hdfv, adj_hdfvs[j]) for j in range(1,
     len(adj_hdfvs))]
        closest_index = distances.index(min(distances))
        unfiltered = merge(unfiltered, masks[closest_index+1], masks[0])
        new_mask = bitwise_or(masks[0], masks[closest_index+1])
        new_hdfv = hdfv(img, new_mask)
        masks.pop(0)
        adj_hdfvs.pop(0)
        masks.pop(closest_index)
        adj_hdfvs.pop(closest_index)
        masks.append(new_mask)
        adj_hdfvs.append(new_hdfv)

    return unfiltered
```

3.3 Classification

3.3.1 Transfer Learning

Transfer learning is a machine learning technique that involves using pre-existing knowledge and models gained from training on one task or domain to accelerate and improve learning in a new context. This technique is used to save time and resources that would have been spent training multiple machine learning models from scratch to complete similar tasks. In traditional machine learning approaches, models are trained and evaluated on a specific task and dataset,

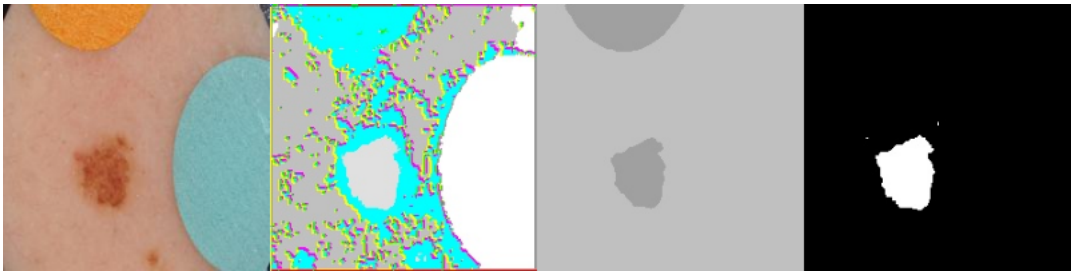

Figure 10: **Stages of Region Merging. LM: Original Image, LC: Oversegmented Image before marginalization, RC: after Region Merging, RM: after Binarizing**

which often requires a large amount of labeled data and computational resources to achieve satisfactory performance. Transfer learning leverages pre-existing knowledge and models to improve learning in a new context, making it a popular approach in deep learning, especially in computer vision and natural language processing tasks. Transfer learning is not its own type of machine learning algorithm. Instead, it's a technique or method that is used during the training of models. To apply transfer learning, different strategies and techniques are used based on the domain of the application, the task at hand, and the availability of data. Transfer learning addresses these limitations by utilizing knowledge learned from a different but related task or domain.

The general process of transfer learning involves the following steps:

(a) Pre-training: A model is trained on a large-scale dataset from a source task or domain. This initial model is typically trained on a task that is related to the target task but may have a different dataset or data distribution.

(b) Feature extraction: The pre-trained model's learned representations or features are extracted from one or more intermediate layers. These features capture valuable patterns and information from the source task or domain.

(c) Fine-tuning: The extracted features are then used as input to a new model, often referred to as the target model. The target model is usually initialized with the pre-trained weights and is fine-tuned on a smaller labeled dataset specific to the target task or domain. During fine-tuning, the target model's weights are adjusted to adapt to the target task while retaining the learned knowledge from the pre-training stage.

The key advantages of transfer learning include:

- Reduced training time and data requirements: By leveraging pre-trained models, transfer learning allows the target model to benefit from prior

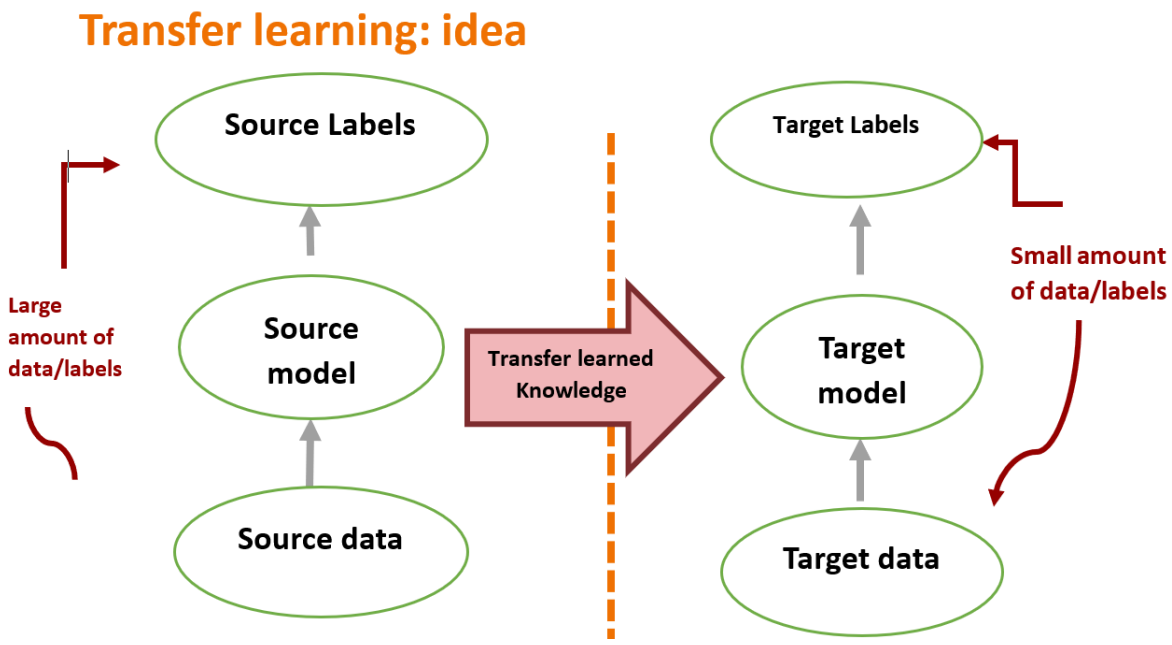

Figure 11: **Transfer Learning**

learning, significantly reducing the amount of labeled data and training time needed for good performance.

- Improved generalization: Pre-training on a large dataset helps the model capture generic features and patterns that are useful across related tasks or domains. This enables the target model to generalize better, even with limited target task data.
- Handling of limited target task data: In scenarios where labeled data for the target task is scarce, transfer learning can provide a more robust solution by leveraging knowledge from a related task with ample labeled data.

Transfer learning has been successfully applied in various domains, including computer vision, natural language processing, and speech recognition. It has contributed to significant performance improvements and breakthroughs in many challenging tasks by effectively transferring knowledge and representations across tasks or domains.

3.3.2 Support Vector Machines (SVM)

Support Vector Machines (SVM) is a machine learning tool that can be used to classify and predict things. It works well with data that has a lot of dimensions and can handle situations where the dividing line between categories is not

straight. SVM tries to find the best line that separates different groups of data. This line is called a hyperplane, and it maximizes the space between the line and the closest data points from each group. These important data points are called support vectors, which is why it's called "Support Vector Machines". To transform the input data into a higher-dimensional feature space, SVM uses a kernel function. The transformed data is then used to find the optimal hyperplane that maximizes the margin. By mapping the data into a higher-dimensional space, SVM can handle non-linear relationships between features.

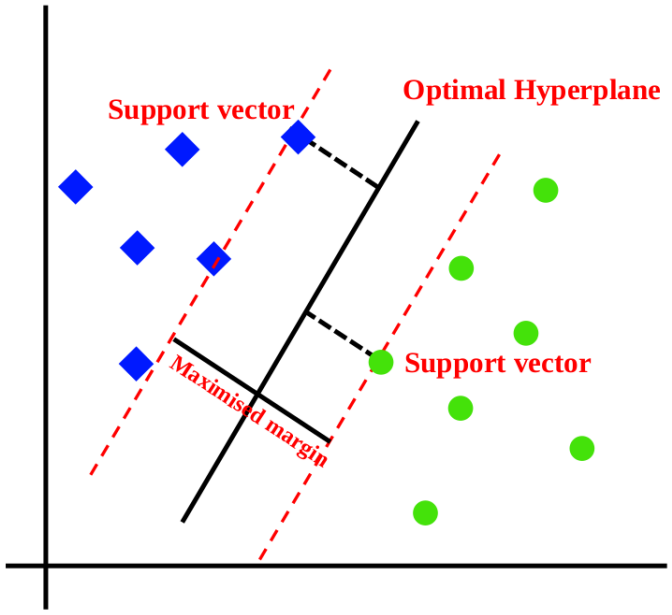

Figure 12: **SVM**

The training process of SVM involves the following steps:

(a) Data Preparation: SVM requires labeled training data, where each data point is associated with a class label.

(b) Feature Transformation: If needed, the input data can be transformed into a higher-dimensional feature space using a kernel function. Common kernel functions include linear, polynomial, radial basis function (RBF), and sigmoid.

(c) Optimization: The SVM algorithm aims to find the hyperplane that maximizes the margin while minimizing the classification error. This optimization problem can be solved using different techniques, such as the Sequential Minimal Optimization (SMO) algorithm.

(d) Model Evaluation: Once the SVM model is trained, it can be used to predict the class labels of new, unseen data points. For classification tasks, the decision is made based on which side of the hyperplane the data point lies.

Table 5: **Typical Optimizers in SVM**

Optimizer	**Description**
Stochastic Gradient Descent (SGD)	Updates model weights using mini-batches of training data. Simple and efficient.
Adam	Uses adaptive learning rates for individual model parameters. Combines AdaGrad and RMSprop. Widely used in deep learning.
RMSprop	Adjusts learning rates based on recent gradient magnitudes. Effective in handling non-stationary data.
AdaGrad	Adapts learning rates based on historical gradient information. Performs well with sparse data.
AdaDelta	Extends AdaGrad to address diminishing learning rate issue. Dynamically adjusts learning rate based on recent gradients.
AdamW	Variant of Adam optimizer with weight decay regularization. Improves generalization performance.

The benefits of SVM include:

- SVM works well when there are many characteristics to consider compared to the number of examples. This makes it a good choice for datasets with lots of dimensions.
- Flexibility with Kernel Functions: SVM allows the use of various kernel functions to handle non-linear relationships between features, enabling it to capture complex patterns in the data.
- Robustness to Outliers: SVM is relatively robust to outliers since the decision boundary is determined by the support vectors, which are the closest data points to the decision boundary.
- Generalization Capability: SVM aims to find the hyperplane with the maximum margin, leading to better generalization performance on unseen data.

SVM has been successfully applied in various domains, including image classification, text categorization, bioinformatics, and finance. It's versatility and strong theoretical foundations make it a popular choice for many machine-learning tasks.

3.3.3 Convolutional Block Attention Mechanism (CBAM)

CBAM (Convolutional Block Attention Module) [22] is a module that enhances the performance of convolutional neural networks (CNNs) by integrating attention mechanisms. It was proposed in the paper by Sanghyun et. Al.[22]. The CBAM module aims to capture both channel-wise and spatial-wise attention in CNNs.

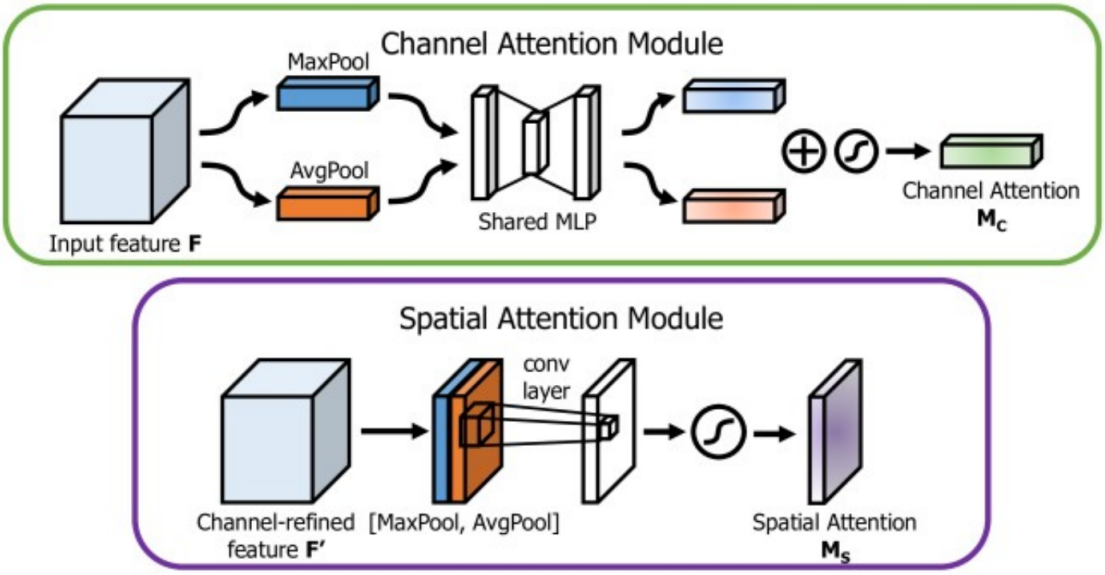

Figure 13: **T: Channel-Attention, and B: Spatial-Attention modules**

It consists of two main components:

(a) Channel Attention Module (CAM): The CAM captures the interdependencies among channels within a convolutional feature map. It computes channel-wise attention weights by considering the importance of each channel. This is achieved through a combination of global average pooling and fully connected layers. The CAM allows the network to emphasize important channels and suppress less relevant ones, enhancing the representation power of the CNN.

(b) Spatial Attention Module (SAM): The SAM captures the interdependencies among spatial locations within a feature map. It computes spatial-wise attention weights by considering the importance of each spatial location. This is achieved through a combination of max pooling and convolutional layers. The SAM enables the network to focus on informative spatial regions while suppressing irrelevant or noisy regions.

By combining the CAM and SAM, the CBAM module generates attention maps that are multiplied element-wise with the original feature maps, allowing the network to adaptively attend to relevant channels and spatial regions at different scales. The benefits of CBAM include:

- Enhanced Representation Learning: The integration of attention mechanisms through the CAM and SAM helps the network focus on important channels and spatial regions, allowing for more informative feature representations.
- Flexibility and Compatibility: CBAM can be easily integrated into existing CNN architectures, as it operates as a modular component that can be inserted into different layers of the network.
- Improved Performance: Experimental results have shown that incorporating CBAM into CNN architectures improves performance on various tasks, including image classification, object detection, and semantic segmentation.

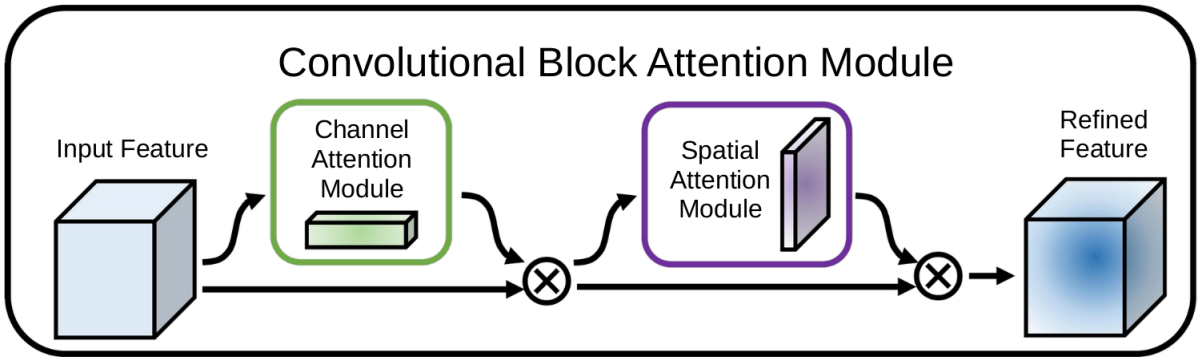

Figure 14: **CBAM framework**

CBAM has demonstrated its effectiveness in improving the representational power and discriminative capabilities of CNNs by capturing channel-wise and spatial-wise dependencies. Its modular design allows for easy integration into existing architectures, making it a valuable tool in computer vision tasks that benefit from attention mechanisms.

4. Chapter: Results and Discussion

4.1 Experimental Setup

The experiments were distributed across multiple platforms: The supervised deep learning tasks were taken up Kaggle cloud instances with NVIDIA P100 accelerator. The unsupervised segmentation tasks were operated using an Intel-i5 1240P 4.4 GHz processor with 16GB of RAM running on Ubuntu 22.04. Both the segmentation tasks were built using Python with Tensorflow-keras as backend. The classification tasks were accomplished in MATLAB R2022a on an Intel-i5 8128U processor with 8GB RAM running on Windows 11.

Table 6: **Optimal Hyperparameters for the DenseNet based U-Net segmentation network**

Backbone	DenseNet201
Loss	bce-dice loss
Optimizer	Adam
Initial Learning Rate	8e-6
Learning Rate scheduler	lr halved if no improvement over 3 epochs
Num. epochs	40
Batch size	4
Input Shape	(224, 224, 3)

4.2 Metrics

This **Supervised Segmentation** framework was tested on both the ISIC 2016 and 2017 datasets where it shows better jaccard and Dice-scores than the best of existing frameworks until 2022. The benchmarks are taken from survey papers, as well as standalone papers, only the highest values are presented for comparision.
For both the segmentation tasks, we are considering the following metrics:

- **Accuracy** or Segmentation Accuracy is an overall measure of how close our segmented outputs are to the ground truth. In case of binary masks, the entire 2-dimensional array is flattened and the TruePositives, FalsePositives, TrueNegatives, and FalseNegatives are computed for the individual pixels in the flattened array. This process will give the accuracy for one

single image, the average accuracy is computed and presented for all the test images in a dataset. Accuracy is defined as:

$$\text{Accuracy} = \frac{\text{True\_Positives} + \text{True\_Negatives}}{\text{True\_Positives} + \text{True\_Negatives} + \text{False\_Positives} + \text{False\_Negatives}} \tag{2}$$

$$\text{Avg\_Accuracy} = \frac{1}{N} * \sum Accuracy_i \tag{3}$$

where N is the number of images

Incidentally, the same formula for accuracy holds true for Classification as well. Individual classes are considered for the True and False values.

- **Precision** is another criteria for judging the performance of any deep learning model. In the context of segmentation, True/False values are computed from the flattened array as the accuracy.

$$\text{Precision} = \frac{\text{True\_Positives}}{\text{True\_Positives} + \text{False\_Positives}} \tag{4}$$

$$\text{Avg\_Precision} = \frac{1}{N} * \sum Precision_i \tag{5}$$

where N is the number of images
Again, the same formula holds for classification tasks as well

- **IoU** or Jaccard_score/index is a criteria specific to segmentation. It is the ratio of the Intersection of positive regions from both ground truth image and generated images, over the Union of the same. This gives us a measure of how much 'deviation' the generated mask shows from the ground truth.

$$\begin{aligned} \text{IoU} &= \frac{\text{Intersection of Predicted and Ground Truth}}{\text{Union of Predicted and Ground Truth}} \\ &= \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives} + \text{False Negatives}} \end{aligned} \tag{6}$$

$$\text{m\_IoU} = \frac{1}{N} * \sum IoU_i \tag{7}$$

where N is the number of images

- **Dice_Score** or F1_Score is another prominent metric used in most segmentation tasks.

$$\text{Dice Score} = \frac{2 \times \text{True Positives}}{2 \times \text{True Positives} + \text{False Positives} + \text{False Negatives}} \tag{8}$$

$$\text{Avg\_Dice} = \frac{1}{N} * \sum Dice_i \tag{9}$$

where N is the number of images

Table 7: **Segmentation Results for the Supervised algorithm on ISIC 2016 dataset**

Method	*IoU*	$F-Score$	*Accuracy*	*Precision*
Tang et. Al.[19]	0.8925	0.9303	0.9716	-
Mohammad et. Al.[13]	0.83	0.91	-	-
Bozorgtabar et. Al.[6]	0.8060	-	-	-
Yuan et. Al.[27]	0.8470	0.9120	0.9550	-
Bi et. Al.[5]	0.8592	0.9177	0.9578	-
Wu et. Al.[23]	0.8692	0.9361	0.9749	-
This method	**0.9086**	**0.9478**	**0.9804**	**0.9709**

Table 8: **Segmentation Results for the Supervised algorithm on ISIC 2017 dataset**

Method	*IoU*	$F-Score$	*Accuracy*	*Precision*
Yuan et. Al.[26]	0.784	-	-	-
Koehrsen et. Al.[12]	-	0.73	-	0.76
Han at. Al.[7]	0.772	0.859	-	-
He et. Al.[8]	0.7580	-	-	-
Mohammed et. Al.[3]	0.7711	0.8708	0.9403	-
Bi et. Al.[5]	0.7773	0.8566	0.9408	-
Tang et. Al.[19]	0.7926	0.8693	0.9431	-
This method	**0.8997**	**0.9435**	**0.9306**	**0.9863**

For the **Fully-Unsupervised segmentation** task, we have generated the masks for all 1279 images in the ISIC 2016 dataset (irrespective of class), and

the metrics that have been computed are as follows:

Table 9: **Metrics for the Unsupervised Segmentation algorithm on the ISIC 2016 dataset**

Average Accuracy:	0.9254582923228345
Average precision:	0.930191650497418
Average Jaccard score (mIoU):	0.7977907233553192
Average Dice score (F1):	0.8759328320199085

There are multiple works done on unsupervised segmentation, however, to the best of our knowledge, there is only one other author Ali et. Al.[1] who has tried a similar fully unsupervised segmentation method on the ISIC 2016 dataset. While their IoU and Dice scores were 0.30 and 0.403 respectively.

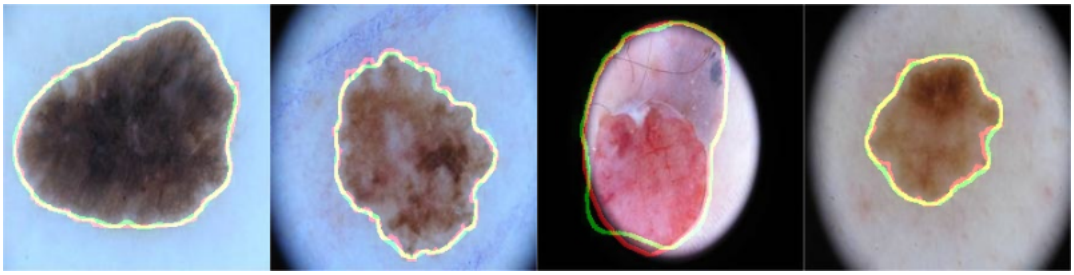

Figure 15: **Some images and their corresponding mask regions as generated by the Supervised Segmentation algorithm.**

Here the red line denotes the Ground Truth region, and the green line denotes the mask region generated by the supervised algorithm.

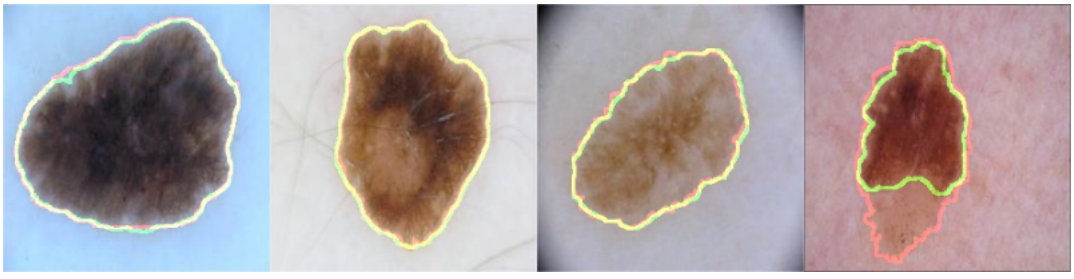

Figure 16: **Some images and their corresponding mask regions as generated by the Unsupervised Segmentation algorithm.**

(above) Here the red line denotes the Ground Truth region, and the green line denotes the mask region generated by the unsupervised algorithm

For the classification task, Transfer Learning based approaches are investigated, features are extracted from the CNN model and the features are sent to SVM classifier for the classification purpose. And apart from this, we worked on CBAM also. CBAM gives better results as compared to the transfer learning model.

Table 10: **Results for the Classification task on ISIC 2016 dataset**

Model	*Accuracy*
Transfer Learning model:	0.8496
Transfer Learning with SVM Classifier:	0.8785
CBAM:	0.9055

5. Chapter: Conclusion and Future Scope

Based on the presented results, we can conclude that these methods, which are built predominantly upon the works of Kanezaki et. Al.[21], Abhishek et. Al.[2], and Khan et. Al.[11], are effective segmentation methods both in the **Supervised** and **Fully-Unsupervised** domain.

Based on this work, we can verify that Illumination based Transformations play a key role in improving the performance of segmentation models, cascaded with other image processing methods such as weighted contrast stretching, Black-Hat filtering, we can create a very effective pipeline for training deep learning segmentation algorithms. Also, it is imperative that the methods can be interchangeably used in both supervised and unsupervised methods. Together, clubbing state-of-the-art deep learning technologies with precise image processing algorithms can give us better and more accurate predictions.

In the future, we are planning to extend this project to incorporate the concept of Object detection and bounding boxes to further enhance the performance of the models. The work would involve accurate localization of skin lesions, which would aid our already existing models to become more robust. Alternatively, we are planning to work on ensembles of multiple models with an endeavour to create a near perfect segmentation algorithm.

We had done Transfer Learning with ResNet-50 and obtained good accuracies. To further improve the results we implemented the SVM classifier, which was found to be the optimal classifier for this task. With the SVM classifier, we obtained better results compared to the previous model. Following that, we also experimented with the Convolutional Block Attention Module, where we achieved even better results, so far being the best. Henceforth, we can conclude that the CBAM cascaded with a Transfer Learning feature extractor, and an SVM Classifier is so far the best approach for the classification task.

In the future, we can develop ensemble methods. Future research could explore the development of ensemble methods that are optimized for real-time applications, which can improve the speed and accuracy of machine-learning models in these domains. Another scope of improvement is to develop multi-scale analysis techniques for medical imaging tasks, Multi-scale analysis can be applied to other medical imaging tasks beyond skin lesion analysis, such as brain imaging, lung imaging, and more. Future research could explore the development of multi-scale analysis techniques that are specifically designed for these applications, which can improve the accuracy and reliability of machine learning models in these domains.

References

[1] Jingpeng Li Abder-Rahman Ali and Thomas Trappenberg. Supervised versus unsupervised deep learning based methods for skin lesion segmentation in dermoscopy images. *Advances in Artificial Intelligence. Canadian AI 2019*, 2019.

[2] Mark S. Drew Abhishek Kumar, Ghassan Hamarneh. Illumination-based transformations improve skin lesion segmentation in dermoscopic images. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2020.

[3] Mohammed A. Al-Masni, Mohammad A. Al-Antari, Min-Te Choi, Seung-Moo Han, and Tae-Seong Kim. Skin lesion segmentation in dermoscopy images via deep full resolution convolutional networks. *Comput Methods Programs Biomed*, 162:221–231, August 2018. ISSN 0169-2607. doi: 10.1016/j.cmpb.2018.05.027. URL `https://doi.org/10.1016/j.cmpb.2018.05.027`.

[4] Polat K Alenezi F, Armghan A. Deep learning based skin lesion segmentation and classification of melanoma using support vector machine (svm). *Asian Pac J Cancer Prev. 2019*, 2019.

[5] Lei Bi, Jinman Kim, Euijoon Ahn, Ashnil Kumar, Dagan Feng, and Michael Fulham. Step-wise integration of deep class-specific learning for dermoscopic image segmentation. *Pattern Recognition*, 85:78–89, 2019. ISSN 0031-3203. doi: https://doi.org/10.1016/j.patcog.2018.08.001. URL `https://www.sciencedirect.com/science/article/pii/S0031320318302772`.

[6] B. Bozorgtabar, S. Sedai, P. K. Roy, and R. Garnavi. Skin lesion segmentation using deep convolution networks guided by local unsupervised learning. *IBM Journal of Research and Development*, 61(4/5):6:1–6:8, 2017. doi: 10.1147/JRD.2017.2708283.

[7] Seung Seog Han, Dong Hyeon Kim, Hyun Jun Kim, Chan Yeong Park, Hyun Soo Kim, and Hyun Ho Kim. Skin lesion classification using convolutional neural network for melanoma recognition. *Journal of Medical Systems*, 42(7):129, 2018.

[8] Xinzi He, Zhen Yu, Tianfu Wang, and Baiying Lei. Skin lesion segmentation via deep refinenet. pages 303–311, 09 2017. ISBN 978-3-319-67557-2. doi: 10.1007/978-3-319-67558-9_35.

[9] Gao Huang, Zhuang Liu, and Kilian Q. Weinberger. Densely connected convolutional networks. *CoRR*, abs/1608.06993, 2016. URL `http://arxiv.org/abs/1608.06993`.

[10] Arslan Javaid, Muhammad Sadiq, and Faraz Akram. Skin cancer classification using image processing and machine learning. *arXiv preprint arXiv:1903.11555*, 2019.

[11] Zubair Khan and Jie Yang. Bottom-up unsupervised image segmentation using fc-dense u-net based deep representation clustering and multidimensional feature fusion based region merging. *Image and Vision Computing*, 94:103871, 2020. ISSN 0262-8856. doi: https://doi.org/10.1016/j.imavis.2020.103871. URL `https://www.sciencedirect.com/science/article/pii/S0262885620300032`.

[12] Will Koehrsen. Transfer learning with convolutional neural networks in pytorch. *Towards Data Science*, 2018.

[13] Pooya Mohammadi, Mohammadreza Babaee, Mohammadreza Hajiabadi, and Mohammadreza Hajiabadi. U-net-based models for skin lesion segmentation: More attention and augmentation, 2022.

[14] Fábio Perez, Cristina Vasconcelos, Sandra Avila, and Eduardo Valle. Data augmentation for skin lesion analysis. In Danail Stoyanov, Zeike Taylor, Duygu Sarikaya, Jonathan McLeod, Miguel Angel González Ballester, Noel C.F. Codella, Anne Martel, Lena Maier-Hein, Anand Malpani, Marco A. Zenati, Sandrine De Ribaupierre, Luo Xiongbiao, Toby Collins, Tobias Reichl, Klaus Drechsler, Marius Erdt, Marius George Linguraru, Cristina Oyarzun Laura, Raj Shekhar, Stefan Wesarg, M. Emre Celebi, Kristin Dana, and Allan Halpern, editors, *OR 2.0 Context-Aware Operating Theaters, Computer Assisted Robotic Endoscopy, Clinical Image-Based Procedures, and Skin Image Analysis*, pages 303–311, Cham, 2018. Springer International Publishing. ISBN 978-3-030-01201-4.

[15] A. Sharif Razavian, H. Azizpour, J. Sullivan, and S. Carlsson. CNN Features Off-the-Shelf: An Astounding Baseline for Recognition. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. DeepVision Workshop*, 2014.

[16] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham, 2015. Springer International Publishing. ISBN 978-3-319-24574-4.

[17] Suresh A Seeja R. A novel multi-task learning network based on melanoma segmentation and classification with skin lesion images. *Diagnostics (Basel). 2023*, 2023.

[18] R Gallagher A Coldman D McLean T Lee, V Ng. Dullrazor: a software approach to hair removal from images. *Comput Biol Med. 1997*, 1997.

[19] Peng Tang, Qiaokang Liang, Xintong Yan, Shuang Xiang, Wei Sun, Dongdong Zhang, and Gianmarco Coppola. Efficient skin lesion segmentation using separable-unet with stochastic weight averaging. *Computer Methods and Programs in Biomedicine*, 178:289–301, 2019.

[20] Manisha Verma and Balasubramanian Raman. Center symmetric local binary co-occurrence pattern for texture, face and bio-medical image retrieval. *Journal of Visual Communication and Image Representation*, 32: 224–236, 2015. ISSN 1047-3203. doi: https://doi.org/10.1016/j.jvcir.2015.08.015. URL `https://www.sciencedirect.com/science/article/pii/S1047320315001583`.

[21] Asako Kanezaki Wonjik Kim and Masayuki Tanaka. Unsupervised learning of image segmentation based on differentiable feature clustering. *IEEE TRANSACTIONS ON IMAGE PROCESSING, Volume 29*, 2020.

[22] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision – ECCV 2018*, pages 3–19, Cham, 2018. Springer International Publishing. ISBN 978-3-030-01234-2.

[23] Xinyu Wu, Yifan Li, Yijun Zhang, Yuxuan Zhang, Yifan Zhang, Yifan Zhang, and Yifan Zhang. W-net and inception residual network for skin lesion segmentation and classification. *Journal of Medical Imaging and Health Informatics*, 11(9):2017–2024, 2021.

[24] Haseeb Younis, Muhammad Hamza Bhatti, and Muhammad Azeem. Classification of skin cancer dermoscopy images using transfer learning. *arXiv preprint arXiv:1904.03143*, 2019.

[25] Z. Yu, X. Jiang, F. Zhou, J. Qin, D. Ni, S. Chen, B. Lei, and T. Wang. Melanoma recognition in dermoscopy images via aggregated deep convolutional features. *IEEE Transactions on Biomedical Engineering*, 66(4): 1006–1016, 2018.

[26] Yading Yuan, Xiaohui Liu, Jagath C Rajapakse, and Xiaogang Wang. Automatic skin lesion segmentation with fully convolutional-deconvolutional networks. *arXiv preprint arXiv:1703.05165*, 2017.

[27] Yixuan Yuan, Min Chao, and Yu-Cheng Lo. Automatic skin lesion segmentation using deep fully convolutional networks with jaccard distance. *IEEE Transactions on Medical Imaging*, 36(9):1876–1886, 2017. ISSN 0278-0062. doi: 10.1109/TMI.2017.2699158.

[28] Ruo Zhang, Ping-Sing Tsai, James Cryer, and Mubarak Shah. Shape from shading: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21:690–706, 08 1999. doi: 10.1109/34.784284.

[29] Tajbakhsh N Liang J. Zhou Z, Siddiquee MMR. Unet++: A nested u-net architecture for medical image segmentation. *Deep Learn Med Image Anal Multimodal Learn Clin Decis Support (2018)*, 2018.

Segmentation and Classification of skin lesion images using deep learning based techniques

ORIGINALITY REPORT

9%	5%	8%	3%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	"Computer Vision – ECCV 2018", Springer Science and Business Media LLC, 2018 Publication	2%
2	Haseeb Younis, Muhammad Hamza Bhatti, Muhammad Azeem. "Classification of Skin Cancer Dermoscopy Images using Transfer Learning", 2019 15th International Conference on Emerging Technologies (ICET), 2019 Publication	1%
3	"Medical Image Computing and Computer Assisted Intervention – MICCAI 2018", Springer Nature America, Inc, 2018 Publication	<1%
4	github.com Internet Source	<1%
5	Rania Ramadan, Saleh Aly. "CU-Net: A New Improved Multi-Input Color U-Net Model for Skin Lesion Semantic Segmentation", IEEE Access, 2022 Publication	<1%