Skin Cancer Classification using Convolutional Neural Network and Feature Selection Strategy

Edosa Tadesse Bayisa

A Thesis Submitted to the department of Computer Science and Engineering,

School of Electrical Engineering and Computing

Presented in Partial Fulfillment of the Requirement for the Degree of

Master's in Computer Science and Engineering

Office of Graduate Studies

Adama Science and Technology University

September, 2024

Adama, Ethiopia

Skin Cancer Classification using Convolutional Neural Network and Feature Selection Strategy

Edosa Tadesse Bayisa

Advisor: Worku Jifara, PhD

A Thesis Submitted to the Department of Computer Science and Engineering

School of Electrical Engineering and Computing

Presented in Partial Fulfillment of the Requirement for the Degree of Master's in Computer Science and Engineering

Office of Graduate Studies

Adama Science and Technology University

September, 2024

Adama, Ethiopia

DECLARATION

This master's thesis, "Skin Cancer Classification using Convolutional Neural Network and Feature Selection Strategy," is my unique creation, I therefore proclaim. In other words, it hasn't been submitted to any other university for the award of a degree, diploma, or certificate. Every source of information used in this thesis has been properly cited and acknowledged.

_____	_____	_____
Name of the student	Signature	Date

RECOMMENDATIONS

This thesis, "Skin Cancer Classification using Convolutional Neural Network and Feature Selection Strategy," was prepared under my/our guidance by Edosa Tadesse and submitted in partial fulfillment of the requirements for the degree of Master of Science. I/we, the advisor(s) of this thesis, hereby certify that I/we have read the revised version of the thesis. Therefore, I/we advocate the submission of amended version of the thesis to the department following the necessary processes.

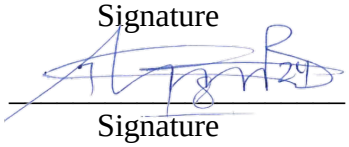
Worku Jifara (PhD)	_____	_____
Major Advisor	Signature	Date
_____	_____	_____
Co-advisor	Signature	Date

APPROVAL

I/we, the advisors of the thesis entitled “Skin Cancer Classification using Convolutional Neural Network and Feature Selection Strategy” and developed by Edosa Tadesse, hereby certify that the recommendation and suggestions made by the board of examiners are appropriately incorporated into the final version of the thesis.

Major Advisor/Supervisor	Signature	Date
Co-advisor/ Co-Supervisor	Signature	Date

To certify that the thesis is accepted for partial fulfillment of the requirement of the degree of Master of Science in Computer Science and Engineering, we, the undersigned, members of the Board of Examiners of the thesis by Edosa Tadesse, have read and evaluated the thesis entitled "Skin Cancer Classification using Convolutional Neural Network and Feature Selection Strategy" and examined the candidate during open defense.

Chairperson	Signature	Date
Internal Examiner	Signature	Date
Asrat Mulatu (Ph.D.)		09/10/2024
External Examiner	Signature	Date

Final approval and acceptance of the thesis is contingent upon submission of its final copy to the Office of Postgraduate Studies (OPGS) through the Department Graduate Council (DGC) and School Graduate Committee (SGC).

Department Head	Signature	Date
School Dean	Signature	Date
Office of Postgraduate Studies, Dean	Signature	Date

ACKNOWLEDGEMENTS

First and foremost, I would like to thank God Almighty for providing me with the determination, strength, and wisdom I needed to complete my thesis.

My profound appreciation goes out to my adviser Worku Jifara (PhD) for all of his support and insightful guidance during my M.Sc. thesis work. He is certainly an inspiration with his patience, motivation, excitement, and vast knowledge.

I am very grateful to my dear wife Rahel Abate and all my children for their long-suffering support in this endeavor and my friend Chala Wakshum for their fantastic support. The success of this research was greatly attributed to their commitment, direction, and readiness to share knowledge. Last but not least, I would want to express my sincere gratitude to all of my friends for their continuous support during my studies.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONOYMS AND ABBREVIATIONS

AKIEC	Actinic keratoses and intraepithelial carcinoma
ANN	Artificial Neural Network
BCC	Basal cell carcinoma
BCC	Basal-cell carcinoma
BKL	Benign lesions of the keratosis
CAD	Computer-Aided Diagnosis
CNN	Convolutional Neural Network
DL	Deep Learning
FN	False Negative
FP	False Positive
FPR	False Positive Rate
FV	Fisher Vector
HAM	Human against Machine
ILSVRC	ImageNet Large Scale Visual Recognition Challenge
ISIC	International Skin Imaging
MEL	Melanoma
Mg	Mammogram
NMSC	Non-melanoma skin cancers
NN	Neural Network
ReLU	Rectifier Linear Unit
RNN	Recurrent Neural Network
SCC	Squamous-cell carcinoma
SGD	Stochastic Gradient Descent

SIFT	Scale-Invariant Feature Transform
SK	Skin cancer
SVM	Support Vector Machine
TN	True Negative
TP	True Positive
US	ULTRASOUND
VASC	Vascular lesions
WHO	World Health Organization

ABSTRACT

The American Cancer Society states that the most common illness among people is skin cancer. It is mainly diagnosed visually, starting with a clinical screening and maybe requiring a histological examination, biopsy, and dermoscopic analysis. Skin cancer is caused by mistakes (mutations) in the DNA of skin cells. In the realm of medical science, identifying a range of local and gene-related illness diseases depends on the ability to classify skin lesions into distinct malignant types. Despite the fact that skin cancers are the worst type of the disease; individuals who are detected with it early on frequently recover completely. Different writers have investigated a number of methods for automatic diagnosis and detection utilizing a variety of techniques. This thesis uses CNN and HOG feature selectors with transfer learning for dermoscopy pictures to demonstrate the classification of skin lesions in those cases where skin cancer is mentioned. This research uses the Kaggle 2018 challenge dataset. This study attempted to use convolutional neural networks to classify photos of skin lesions. Deep neural networks demonstrate enormous promise for classifying images while accounting for the significant environmental heterogeneity. The mutations cause the cells to grow out of control, resulting in a mass of malignant cells. The division of these lesions into various malignant types, including benign keratosis (BKL), actinic keratosis (AKIEC), basal cell carcinoma (BCC), melanoma (MEL), melanomic neves (NV), and vascular lesions (VASC), offers some insight into the problem. The process utilized in this work produces the best results when we mix the CNN model, HOG feature selector, and transfer learning. The accuracy we obtain is 96.0%.

Keyword: *skin cancer classification, CNN, HOG feature selector, Transfer learning, melanoma, dermatology.*

CHAPTER ONE

INTRODUCTION

This chapter provides an explanation of the study's background, objectives, scope, limitations, and organizational structure. It also describes the issues, research questions, and motivation for the investigation.

1.1. Background of the study

One of the biggest causes of death among people is cancer. The World Health Organization predicts that by 2030, cancer would account for the majority of deaths (13.1 million) (World Health Organization, 2018), (Aswir & Misbah, 2018). The most prevalent type of cancer in the USA is skin cancer, out of all the different types (Coates et al., 2011). According to projections, 20% of Americans will get skin cancer at some point in their lives (Johansson et al., 2019).

Death from skin cancer is not a guaranteed outcome. However, early detection is essential to preserving life. Understanding the early detection and identification of skin cancer requires a thorough examination of human skin and the numerous forms of skin malignancies.

This chapter is divided into three parts. The first section covers the layers of human skin, the second section explains the various forms of skin cancers, and the third section goes into more detail on computer-aided diagnosis methods.

Based on the most recent WHO data released in 2020, 1,006 skin cancer deaths in Ethiopia, or 0.18% of all fatalities, were reported. Ethiopia's age-adjusted death rate of 1.82 per 100,000 people places it 97th in the world.

Human Skin Layers

At 1.5 to 2.0 square meters on average, the skin is the biggest organ in the human body. It shields the body from UV radiation and diseases (Kuppam, 2020), and synthesizes vitamin D. Skin is comprised of three main layers: the epidermis, the dermis, and the hypodermis.

The epidermis, or top layer of human skin, is made up of stratified squamous cells and basal lamina. Basal cells are the spherical, flat skin cells that lie beneath squamous cells. The epidermis lacks blood arteries, and diffusion enables oxygen to enter the cells' lowest layer. Skin color is attributed to melanin, a pigment found in the deepest layer of the epidermis.

Dermis: It is made up of various cell types that form blood arteries and sweat glands. The dermis protects the body from strain and pressure by functioning as a cushion.

Hypodermis: The hypodermis is not always thought of as a skin layer, despite being a recognized skin component. It links the skin to the muscles and bones and is located beneath the dermis. Adipose (fat) and connective tissue are found in the hypodermis. The cellular building blocks and layers of these layers are depicted in Figure 1.

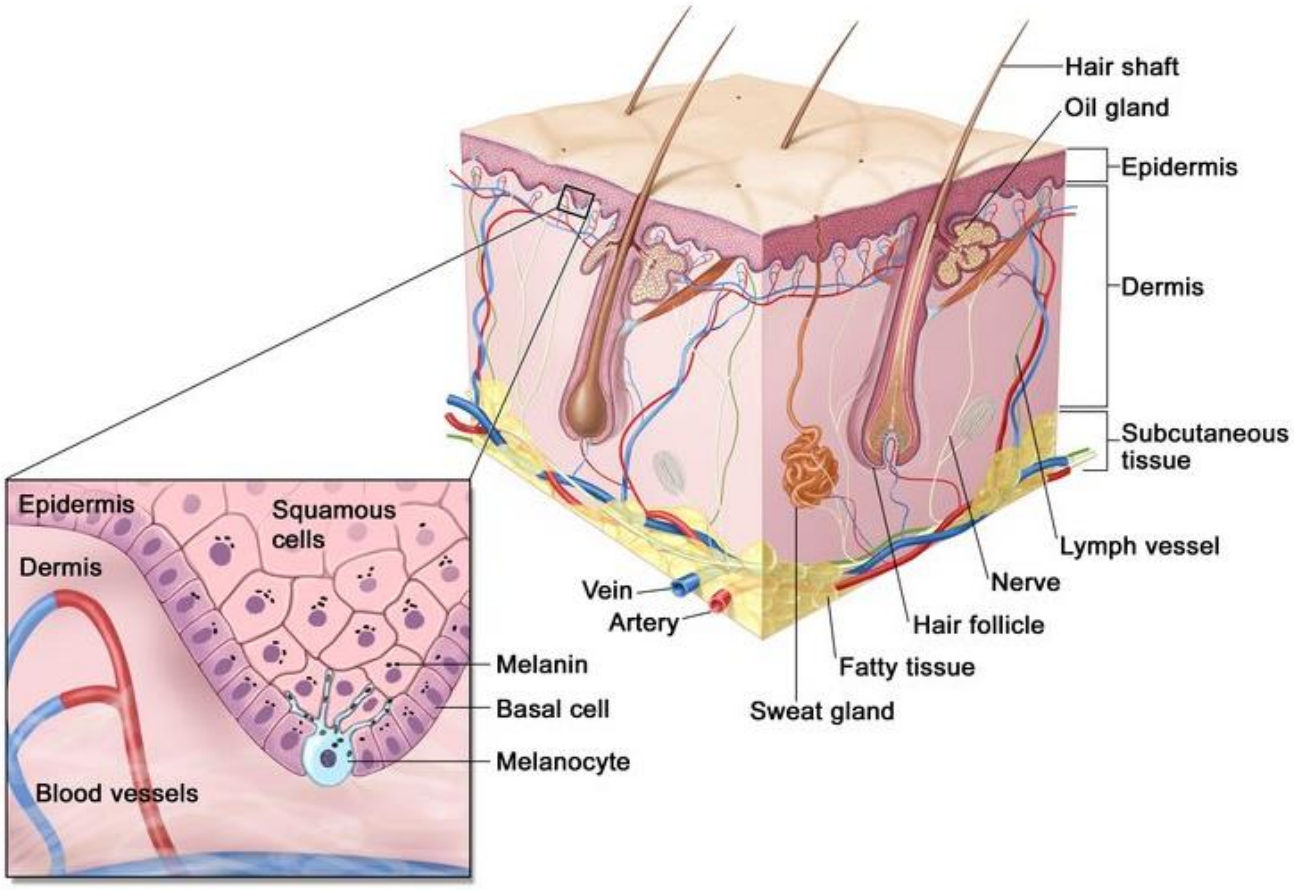


Figure 1. 1: The layers of human skin called the epidermis and dermis

1.2. Skin Cancer

The living cells that make up the human body divide, expand, and eventually perish. The human body uses cell division, a continuous process, to replace dying cells. Nonetheless, cancer is caused by aberrant cell growth and uncontrollably fast cell division (Beger et al., 2008).

One of the most prevalent types of cancer in people is skin cancer, which develops on the skin as a result of aberrant cell growth that can quickly infiltrate and spread to other parts of the body (Aswir & Misbah, 2018). Skin malignancies fall into three primary categories: (1) Squamous-cell carcinoma (SCC), (2) Basal-cell carcinoma (BCC), and (3) Malignant melanoma is the first. Two forms of non-melanoma skin cancers (NMSC) are BCC and SCC.

One of the most deadly and quickly spreading cancer kinds worldwide is melanoma. Every year, 3.5 million cases of skin cancer are identified in the USA. With the exception of melanoma, the sixth most frequent type of cancer in the United States, skin cancer is rarely lethal (Erol et al., 2017). The group of women most typically affected by melanoma is those aged 25 to 29. The World Health Organization has classified ultraviolet tanning devices with plutonium and cigarettes as known or probable human carcinogens (Erol et al., 2017). In the United States, melanoma claimed the lives of an estimated 87,110 adults in 2017. Of those, about 9,730 died from the disease. Melanoma is primarily caused by UV radiation exposure that damages DNA (i.e., sun light and tanning beds). Fair skin type and genetics with a history of malignant melanoma are the additional risk factor (Aswir & Misbah, 2018), (Aswir & Misbah, 2018). Melanoma is the name for a cancer of the melanocytes. Melanocytes are special skin cells that are located in the epidermis' outer layer. Because melanoma forms in the epidermis, it is apparent to the unaided eye. Early identification and treatment are crucial to preventing injury. If melanoma is discovered early, it can be treated with an excision operation. But the largest barrier to early treatment is the high rate of false-negatives for malignant melanoma (Walter et al., 2013). Melanoma is commonly found on the lower limbs in female patients and on the back in male patients (World Health Organization, 2014), however, it is also quite uncommon to find it on other organs with cells, such the mouth and eye (Aswir & Misbah, 2018). Malignancy of basal cells The most common type of skin cancer is basal-cell carcinoma (BCC), with at least 4 million cases reported to the US authorities annually. It appears from the lowest layer of the epidermis. Generally speaking, BCCs resemble red spots or open sores (Figure 2). BCCs nearly never spread, thus instances of them

doing so are quite rare (Aswir & Misbah, 2018), but people who have had BCCs are prone to develop it again in their lifetime.

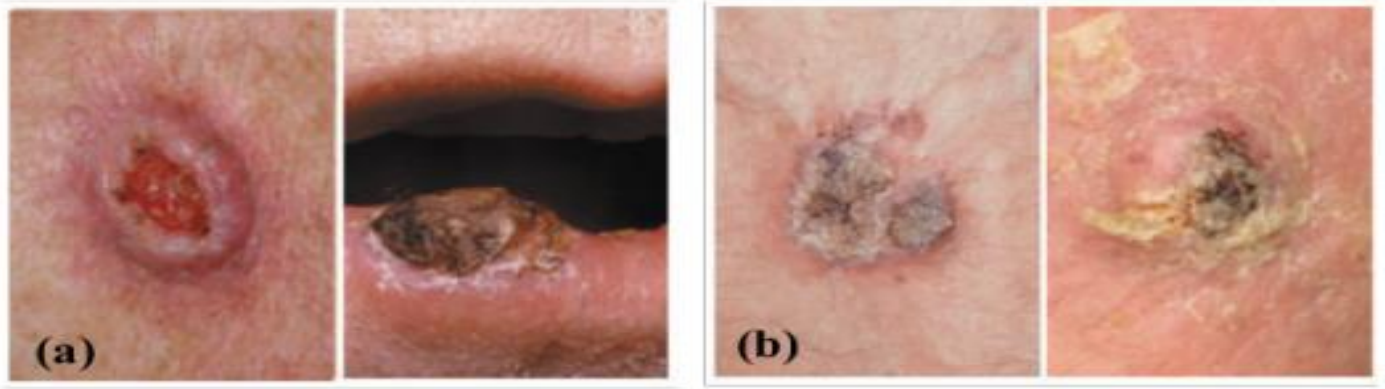


Figure 1. 2 Types of basal-cell carcinoma include an open sore (b) and a red patch (a).

Squamous-Cell Carcinoma (SCC): frequently starts off as a little bump, grows over time, and eventually develops into an ulcer. SCCs are more likely to migrate to other organs and have more asymmetrical forms and crusted surfaces than BCCs (Dermatology.Pdf, n.d.). Along with UV exposure, immune suppression is a significant risk factor for SCC.

Melanocytes: These cells, which are found in the bottom layer of the epidermis, are melanin-producing. The pigment known as melanin is responsible for the color of skin. Melanocytes that are exposed to sunlight produce more pigment, which causes skin to darken. The two most common types of skin cancer are basal cell carcinomas and squamous cell carcinomas. They begin in the basal and squamous layers of the skin, respectively. Both can usually be cured, although the process of doing so can be expensive and deformity inducing. Melanoma, the third most common type of skin cancer, begins in melanocytes. Because melanoma is more likely to spread to other body regions, particularly vital organs, it is the most lethal kind of skin cancer.

The choice of treatment should be chosen by the histological type of the lesion its size and location and the age of the patient. Not every lesion responds well to the same kind of treatment. The objectives of treatment are complete tumor eradication, function preservation, and good esthetic results outcome (Silpa & V, 2013).

In order to mitigate the risk of skin cancer, it is imperative that individuals receive comprehensive education about the disease. This will allow for the possibility of preventing skin cancer to some

degree. Some preventive measures include avoiding direct sunlight during the hottest part of the day, wearing or applying sunscreen when exposed to sunlight, and avoiding sun lump exposure (Silpa & V, 2013).

1.3. Motivation for the study

Health is undeniably important in life and ensuring the safety of humans is a critical duty, that is why extensive discoveries and technologies have been investigated in this area ever since the beginning of human history. Even though these advancements bring a lot of achievement in the sector, some of them have undeniable side effects. Therefore, lowering dangers associated with diagnostic techniques that make use of advanced technology is another concern nowadays.

According to the research articles I've read on the subject, skin cancer claims many lives. This issue affects a great deal of people all across the world. The death toll is rising daily and the issue is quite serious. But when we get back to the African continent, this issue is killing a lot of people. In a similar vein, the illness does occasionally, though infrequently, strike our nation. Accurate diagnosis is not sufficient when it comes to diagnosis. I wanted to contribute to the research on skin cancer, which is why I was driven to perform this study.

1.4. Statement of the Problem

Many studies are being conducted to classify skin cancer automatically in an effort to address the issue of accuracy in CAD of skin lesions (Xu et al., 2020). Recent studies of automatic skin lesion classification towards skin cancer show that, the accuracy of Skin lesion segmentation which is the first step in CAD affects classification of skin cancer (Xu et al., 2020), (Ghosh et al., 2019a), (Yildirim & Çinar, 2021), (Ghosh et al., 2019a). The segmentation completely determines the outcome of the automatic examination of the skin lesion (De Benedetto et al., 2012), result because features for skin lesion classification are selected from segmented region of image.

Based on the stated CAD and clinical diagnosis problems, this research proposes classification of skin cancer using CNN model with feature selection strategy from public dermoscopic skin cancer images. When there is a smooth transition between normal and lesion skin, the problem of skin lesion classification around the boundaries of the lesion is solved by integrating CNN with feature selection strategy algorithms for skin lesion segmentation.

The work aims to solve the aforementioned difficulties by building a CNN model that classifies photographs of skin cancer from the public domain through the use of a feature selection technique.

1.5. Research Question

The following three significant research questions are what this study seeks to address.

RQ1. To what extent feature selection strategy can help the improvement of deep learning model performance?

RQ2. How to design a deep learning model with feature selection strategy for the skin cancer classification?

RQ3. Can we improve skin cancer classification performance via transfer learning and feature selection strategy?

1.6. Objective of the Study

1.6.1. General Objective

The general objective of this study is to classify skin cancer by **using CNN model with the integration of feature selection** method.

1.6.2. Specific Objective

In order to accomplish the overall goal, the following particular goals will be addressed:

1. Study skin cancer classification literatures to understand the most related works.
2. Study the benefit of using feature selection algorithm and transfer learning.
3. Design a deep learning model with the integration of feature selection strategy and transfer learning for skin cancer classification.
4. Evaluating the performance of the skin cancer classification via the proposed feature selection and transfer learning techniques.

1.7. Scope and Limitation of the Study

1.7.1. Scope of the Study

The goal of this research is to create a feature selection approach for the CNN model that will help classify skin cancer. The study's objective is to classify skin cancer using deep CNN in conjunction

with a feature selection strategy of filtering approach to provide the radiologist with a report that is more accurate.

1.7.2. Limitation of the Study

This study does not include imaging modalities other than Dermatology.

1.8. Significance of the Study

This work could be one of the contributions in the area of classifying the skin cancer images with some enhancements and full integration of the suggested model. So that the radiologist and/or physicians may get a better diagnosis and make better decisions while the patient's exposure to radiation malignant or not. Furthermore, the work can be a valuable resource for scientists conducting additional research and investigation in this field.

1.9. Research methodology

The stages and processes utilized to carry out the research are described in the research methodology, along with a list of the instruments required. We specifically used an experimental form of research methodology in this study. The suggested study uses feature selection techniques in conjunction with a CNN model to classify skin cancer cases.

1.10. Literature Review

In this specific study, identifying and understanding theoretical concepts and associated research are crucial steps. A thorough review of the literature is done in order to fully comprehend the research area. Additionally, knowing the instruments used for experimentation is crucial to finishing the research properly. Because of this, the literature review that was done for this study focuses mostly on important issues associated with these components.

Data collection technique

Collecting data for skin cancer classification typically involves acquiring a dataset of images that are labeled with information about whether the skin lesion is benign or malignant. Here are some techniques and considerations for collecting a dataset for skin cancer classification.

Algorithms Used

A range of machine learning techniques, including neural networks, artificial neural networks (ANN), convolutional neural networks (CNN), and other image classification-based algorithms, will be used to

identify and categorize photo collections. The performance of these algorithms will be assessed and contrasted with one another. The ultimate algorithm for the categorization of skin cancer will then be selected based on which algorithm produces the most precise and effective results (Wadajo & Science, 2023).

Tools

The selection of appropriate instruments, which can help with both planning and implementing the architecture, is crucial to the research's success.

Anaconda: is a free and open-source computer vision toolkit written in Python and R that is mostly used in data science and machine learning. It is utilized for the model's implementation. It offers a feature-rich package management system together with an assortment of pre-built tools and libraries that are frequently utilized in data analysis and scientific computing. Installing, updating, and switching between different packages and dependencies is made simpler for users by Anaconda, which speeds up the process of setting up and managing software environments for tasks involving data. Development environments such as Jupyter Notebook are included in Anaconda, along with popular data science libraries like NumPy, Pandas, Matplotlib, and SciPy. It also includes Anaconda Navigator, a graphical user interface that simplifies package management, app opening, and environment setup. An open-source machine learning framework called Tensor Flow is frequently used, especially in deep learning applications, to create and implement machine learning models. The ability to work with computation graphs is one of its primary features. Tensor flow's architecture functions for data preprocessing, model construction, model training, and model estimation.

Tensor (an n-dimension array) is used in all computations in Tensor Flow and is used to represent many types of data. Tensor Flow uses a graph structure to visually depict the training series of computations. It has two different CPU and GPU distributions. It supports a wide range of applications, including common machine learning techniques like decision trees, support vector machines (SVMs), and linear regression, in addition to deep learning. Additionally, it offers libraries and tools for model serving, data visualization, and preprocessing. Keras, a high-level API provided by TensorFlow, makes neural network construction and training easier (Wadajo & Science, 2023).

Keras: Built on top of Tensor Flow and other deep learning frameworks, Keras is an application programming interface (API) for high-level neural networks. It provides an easy-to-use interface for developing, optimizing, and applying deep learning models. The model is highly expandable using

Python, easy to use, and most importantly, comes with pre-trained CNN models. For model training, assessment, and inference, it offers a variety of tools and features. It has integrated techniques for preparing data, augmenting data, and visualizing models.

1.11. Organization of the Thesis

The six chapters that make up this thesis work are broadly defined as follows. Chapter one: provides an outline of the thesis work and provides arguments for the methods and procedures that will be followed.

Chapter two: The literature and relevant works on image-to-image classification for skin cancer are covered in this chapter.

Chapter three: This chapter attempts to provide a clear explanation of the research approach, including everything from the methods for gathering data to the metrics used to evaluate the model.

Chapter four: This chapter attempts to provide comprehensive details on the ways in which the suggested solution attempts to address the issues.

Chapter five: This chapter presents a comparative analysis of the results using figures, graphs, and tables while also discussing the outcomes of the suggested solution.

Chapter six: The research's general overview is provided in this chapter. provides the work's conclusion as well as potential next directions.

CHAPTER TWO

Literature Review and Related Works

2.1. Introduction

Convolutional Neural Networks (or CNNs or ConvNets) are a unique kind of multilayer neural network for spatial data among many deep learning (Ghosh et al., 2019b) architectures. The way that living things see has influenced CNN's architectural design. Although it gained popularity following AlexNet's 2012 record-breaking performance (Gonzalez, 2007), it was first used in 1980. Following 2012, CNN began to dominate a number of computer vision and natural language processing domains, among others. Skin cancer is a disorder where malignant cells grow in the skin tissues. Still, skin cancer is a serious illness, and early discovery facilitates quicker treatment. The current work describes an autonomous computer-aided method for early skin cancer diagnosis (Xu et al., 2020).

Cells can grow and divide quickly in some cancer forms while more slowly in others. While some cancers, such as leukemia, do not cause obvious growths known as tumors. The word "cancer" describes a group of diseases that can impact any part of the body. Malignant tumors of the brain, lung, skin, prostate, skin, and colon are among the diseases known as cancer.

One type of skin cancer that causes malignant tumors on the skin is melanoma. Dermatological pictures are used to find skin cancer. Skin cancer classification using machine learning with high-performance images has a decent identification rate (Hasan et al., 2019). Yet, by extracting additional features, the model's accuracy can be improved while its sensitivity is more skewed. In (M. S. Ali et al., 2021a), the author put out a technique that makes use of image processing processes to improve the accuracy of skin cancer diagnosis (M. S. Ali et al., 2021b).

2.2. Computer Vision:

Algorithms for computer vision are not magical; in order to function, they require data, and the quality of the data they receive depends on it. Since large data sets offer a multitude of entities, relationships, and clusters for the learning process, they are essential for the development of computer vision algorithms. The algorithm requires data on a variety of business processes from a variety of sources in a variety of formats in order to expand and improve the correlations it makes.

2.3. Concept of computer vision and skin cancer classification.

Deep Learning is a specialized skill set that has seen tremendous demand in the IT industry in recent years. This ability, together with developments in ML and AI, has significantly improved the speed and effectiveness of data processing in data science. Computer Learning, a subclass In AI, concentrates on giving computers the ability to learn and carry out classification and decision making tasks without being specifically programmed. Deep Learning, a branch of machine learning, uses multi-layered neural networks to comprehend intricate patterns and their connections in data. Deep learning has revolutionized several technological fields in the modern era, with computer vision being one of the most prominent examples. The amazing ability of computers to independently analyze and interpret images and movies is known as computer vision. It stands as one of the most prominent and actively discussed subjects in the industry (Wadajo & Science, 2023). As a result, computer vision is essential to the operation of technologies like robotics, self-driving automobiles, and biometrics used in the classification of skin cancer. But image processing lies at the heart of computer vision (Gutema & Engidawork, 2018). The field of computer vision is concerned with creating instruments and ideas that enable the development of machines that can extract useful information from visual input, such as photographs and movies. Using these methods to interpret the world is its main goal.

2.4. Neural Network (NN)

It's a kind of machine learning model that draws inspiration from the composition and operations of the human brain. Layer-by-layer organization of interconnected nodes, or neurons, makes up neural networks. These layers carry out the information flow, and each neuron uses its inputs and an activation function to carry out a basic computation. Neural networks consist of multiple layers (input, hidden, and output), with varying numbers of neurons or nodes. These layers have trainable weights and biases that carry out mathematical operations to identify patterns in data and generate output. Every neuron has complete connectivity to every other neuron in the post- and preceding layers. The NN's central processing unit consists of these neurons. A hidden layer sits between the input and output layers, processing and passing the'message' from the previous layer to the post layer. Each neuron receives some inputs from neurons in the previous layer. The first layer is the input layer, which was viewed as a single vector that uses for accepting the data and passes to the other networks. The last 20 layers are the output layer, whose output view is as predict result (Wadajo & Science,

2023). The output of this neuron optionally functions as a non-linear function after it has completed the dot product of all the inputs. These following algorithms are explained in the context of neural networks, which are relevant to our research. Every one of these models has advantages and disadvantages, and they are appropriate for certain kinds of jobs and data.

2.5. Artificial Neural Networks:

This form of machine learning model was developed with inspiration from the organic neural networks seen in the human brain. (ANN) is a machine learning model in image processing that is used for tasks including object identification, picture segmentation, and image classification. The output layer generates the required output, such as categorizing the image, once the input layer gets the image data. The raw data is processed and transformed by hidden layers in between. Through training, ANNs modify their weights and biases to identify patterns and features in images. After being trained, ANNs are capable of identifying objects and particular features in pictures. ANNs are commonly used for problems related to pattern recognition, regression, and classification. They are inspired by biological neural networks. ANNs use a training procedure in which the network modifies its weights and biases to identify patterns and features in the images.

2.6. Supper Vector Machine (SVM)

This approach for supervised machine learning is employed in tasks related to regression or classification. SVMs may handle datasets that are linearly separable or non-linearly separable, and they perform well with high-dimensional data. SVMs are used in classification with the goal of identifying the best hyperplane to divide classes while optimizing the margin between them. Support vectors are those data points that are closest to the decision border, and they are identified. Better class separation is possible with SVMs by using kernel functions to convert input data into a higher dimensional feature space. SVMs work well in image processing for problems including segmentation, object recognition, and picture classification. They categorize photos into distinct groups by using discriminative features they have learned from the images. SVMs use kernel functions to improve class reparability and are particularly good at handling high dimensional feature spaces.

2.7. Convolutional Neural Networks(CNN):

In machine learning (ML), patterns are discovered by running certain algorithms over vast amounts of historical data until the system can accurately predict fresh data. One kind of deep artificial neural

network that has demonstrated success in machine learning (ML) for the analysis of visual input is the convolutional neural network (CNN). Among the many uses for CNNs is image categorization. CNN belongs to the category of Deep Neural Networks that excel at identifying and categorizing specific features within images (Wadajo & Science, 2023).

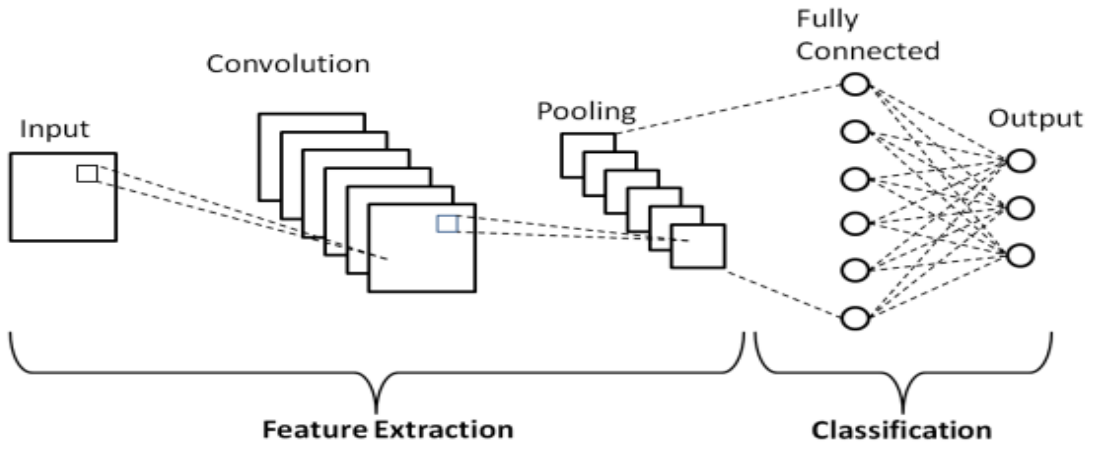


Figure 2. 1 An illustration showing the fundamental architecture of a convolutional neural network (CNN)

2.7.1. Convolution Layers (CONV)

A convolution layer is a fundamental component of the CNN architecture that performs feature extraction, which typically forward propagation on a training dataset, and learnable parameters, i.e., kernels and weights, are updated according to the loss value through back propagation with gradient descent optimization algorithm. ReLU, rectified linear unit consists of a combination of linear and nonlinear operations, i. e., convolution operation and activation function (Patil & Rane, 2021).

2.7.2. Pooling Layer/POOL

A pooling layer provides a typical down sampling operation which reduces the in-plane dimensionality of the feature maps in order to introduce translation invariance to small shifts and distortions, and decrease the number of subsequent learnable parameters. It is of note that there is no learnable parameter in any of the pooling layers, whereas filter size, stride, and padding are hyper parameters in pooling operations, similar to convolution operations (Patil & Rane, 2021).

2.7.3. Fully connected (dense) layer/FC

The output feature maps of the final convolution or pooling layer is typically flattened, i.e., transformed into a one-dimensional (1D) array of numbers (or vector), and connected to one or more fully connected layers, also known as dense layers, in which every input is connected to every output by a learnable weight. Once the features extracted by the convolution layers and down sampled by the pooling layers are created, they are mapped by a subset of fully connected layers to the final outputs of the network, such as the probabilities for each class in classification tasks. The final fully connected layer typically has the same number of output nodes as the number of classes. Each fully connected layer is followed by a nonlinear function, such as ReLU, as described above (Patil & Rane, 2021).

Convolutional operation- Every convolutional layer undergoes a convolutional operation. Between the filter and the image section, there is an element-by-element multiply-sum operation. Focuses on deleting or keeping important elements—like the image—from the input. It creates the reduced image or feature map.

Mathematical operation:

Enter: N*N

F*F filter

Then, (N-F+1)*(N-F+1) = (N*N) * (F*F) => Convolutional Operation: An application of filters or kernels

Pooling operation: Occurs b/n a section of feature map and filter. It is also called sub-sampling, used to reduce the dimensionality of feature maps from the convolution operation (Wadajo & Science, 2023). Is employed in the process to deal with various types of distortions or to preserve spatial invariance. A pooled feature map is the output. Max pooling returns the highest value from the region of the image that the kernel or filter covers, while Average Pooling returns the average of all the values in the region that the kernel or filter covers. These two processes are used.

Mathematical operation using the H*W*C feature as a consideration:

Where: H stands for the feature map's height, W for its width,

The feature map's C channel, the filter F, and the stride length S

Next, [(H-F)/S] +1*[(W-F)/S] +1*C = outcome of the pooling process

The final pooled feature map, or output from the last pooling layer, is fed into a Multi-Layer Perceptron (MLP) in a CNN to be classified into a certain class label. This process is known as the flatten operation. Since the MLP can only handle one-dimensional input, a single column is created from the final pooled feature map. The MLP uses this flattened representation as its input data. The pooled feature map is converted into a vector during this process, serving as the input layer for further neural networks.

Fully connected operation- The entire recognition and classification process is finished by this fully integrated operation. The two main functions of the CNN algorithm are convolution and sampling, which happen on the convolutional layers and max pooling layers, respectively.

2.8. CNN Application

CNN is used to address computer vision tasks like object identification, picture recognition, image segmentation, image classification, and other ML-based issues. However, CNN requires a massive volume of data in order to train. When there is enough labeled data available for training, CNN has primarily shown impressive results in traffic sign detection, medical picture segmentation, skin cancer, and object identification in natural photographs (Teuwen & Moriakov, 2019).

2.9. Some Applications Areas of CNNs

This section covers some of the main use cases for CNN that use state-of-the-art performance, such as object detection, picture captioning, action recognition, text restructuring, image categorization, and human position estimation, etc (Ghosh et al., 2019b).

2.9.1. Image Classification

CNN has improved classification accuracy due to many features including weight sharing and several levels of feature extraction, such as classifiers compared to other methods especially in the case of large scale datasets (Sultana et al., 2018). The creation of AlexNet in 2012, which took first place in the ILSVRC challenge that same year, represents the first significant advance in picture categorization (Russakovsky et al., 2015). Subsequently, the CNN model has undergone multiple iterations by researchers, solidifying its position as the preferred solution for image classification problems.

2.9.2. Medical Image Analysis

The rapid advancement of CNN-based image analysis has demonstrated CNN's state-of-the-art basis, as seen by its better performance in illness diagnosis through the processing of medical images such as MRIs and X-rays (Anwar et al., 2018). These days, a wide range of medical conditions, including diabetes, Parkinson's disease, brain tumors, pneumonia, breast cancer, and many more, may be accurately diagnosed using CNN-based models.

2.9.3. Text Recognition

Text identification and detection inside images have long been the subject of extensive research. CNN made its initial significant contribution to this field with LeNet-5, which accurately identified the data in the MNIST dataset (Deng, 2012). After that, CNN has made a number of advancements in recent years and now plays a crucial function (Wang et al., n.d.). To identify the text included within the image, which consists of letters, numbers, and symbols from several languages.

2.9.4. Image Caption Generation

It refers to acquiring a description of the target image, encompassing the identification and detection of various items within the image together with a status description. Here, we completed the first task using CNN and created a textual status description using a number of Natural Language Processing (NLP) approaches (Karpathy, 2015).

2.9.5. Action Recognition

Based on the visual appearance and motion dynamics of any particular human body, a number of effective CNN base techniques can currently predict the actions or behaviors of human subjects with high accuracy. In terms of AI, this propels CNN to new heights. It involves identifying motion from still photos or from a video clip.

CNN is a popular picture classification tool. CNN is a great tool for using histology images to diagnose cancer. Using a dataset of manually created descriptors, it was utilized to diagnose skin cancer from images and compare the outcomes to a pre-trained network. Data augmentation is used in the second stage to solve the problem of class skewness. For the purpose of classifying images, pre trained networks are readily accessible. Image categorization might be easy if the needed image can be found in a labeled dataset.

2.10. Model Selection

One of the most well-known CNN architectures is VGGNet, which Simonyan and Zisserman unveiled in 2014. Out of the six CNN configurations that the authors presented, the most effective ones are VGGNet-16 (configuration D) and VGGNet-19 (configuration E) (Simonyan & Zisserman, 2015).

As we describes above we select the VGG16 model and the HOG features as feature selector and the Transfer learning as pertained model for our research in order to image classification or dataset preprocessing.

Among the top-performing architectures in the 2014 ILSVRC challenge was VGG-16. With a top-5 classification error of 7.32%, it was only surpassed by GoogLeNet, which had a classification error of 6.66%, to finish second in the classification task. It was the task winner as well, with a localization error of 25.32%.

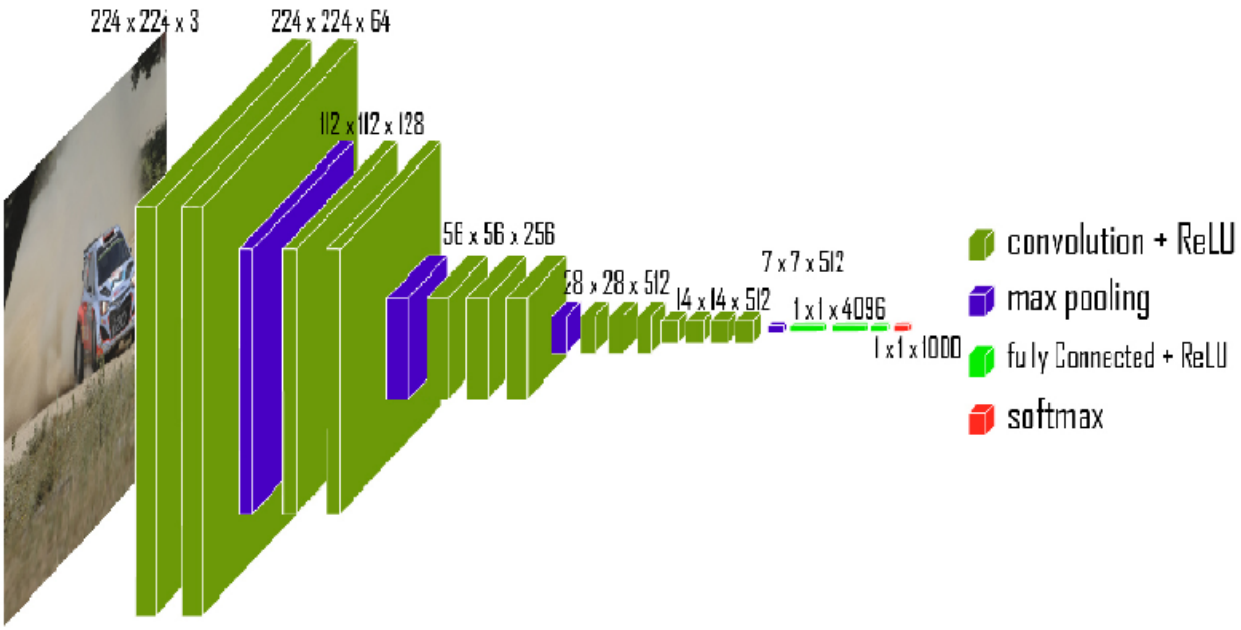


Figure 2. 2VGG-16 architecture

One of the main advantages of VGG16 is its deep architecture, which allows it to learn more complex features from the input image. The use of small kernel sizes (3x3) in the convolutional layers also helps to reduce the number of parameters in the network, which makes it easier to train.

Histogram of Oriented Gradient (HOG) Feature selector: The dimensionality of the data used for data mining and machine learning applications has skyrocketed over the last thirty years. The curse of dimensionality, or data with exceptionally high dimensionality, has posed significant hurdles to current learning techniques (Aggarwal, 2014). A learning model's performance degenerates as a result of overfitting caused by an abundance of features. Dimensionality reduction strategies have been researched as a means of addressing the issue of the curse of dimensionality. In the fields of data mining and machine learning, this study area is important. Feature selection is one technique that practitioners commonly employ to decrease dimensionality. Its objective is to use a certain relevance assessment criterion to choose a subset of relevant features from the original collection. Better interpretability of the model, lower processing costs, and enhanced learning performance (such as increased classification learning accuracy) are typically the outcomes of this (Miao & Niu, 2016).

Transfer Learning with VGG16: Applying a previously learnt model to a new problem is known as transfer learning, which is a deep learning technique. Because VGG16 has been pre-trained on Image Net, which has over a million photos and 1000 classes, it is a well-liked model for transfer learning. We may leverage the learnt features and refine the model on a smaller dataset for a new challenge by starting with VGG16. A machine learning technique called transfer learning involves applying a model that has been trained on one task to another that is similar. Models are trained separately for each unique job in traditional machine learning. Transfer learning, on the other hand, enables models to apply previously learned information from a source task to enhance performance on a target task.

2.11. Related work

We will try to review researches which done on the areas of skin cancer. Artificial intelligence and dermatologists were viewed as rivals in these investigations. Nonetheless, the amalgamation of classifiers often produces better outcomes, in both machine learning and human evaluation. The purpose of this study was to determine whether merging artificial and human intelligence may improve the classification of skin cancer (Hekler et al., 2019).

This study has proposed convolutional neural network based techniques for melanoma classification.

A method is created to assist medical professionals and patients in determining the type of skin cancer—malignant or benign.

According to the experimental and assessment part, the model can help medical practitioners identify skin cancer by serving as a standard (Hekler et al., 2019). Any doctor can obtain accurate results by taking a few random photos, but with the traditional method, it takes too long to identify cases accurately (Centers for Disease Control and Prevention, 2014).

The thesis on "Classification of Skin Cancer Images using Convolutional Neural Networks" aims to develop an effective model for classifying skin cancer types, particularly melanoma and non-melanoma, using deep learning techniques. The study highlights significant challenges such as varying image resolutions, class imbalances in datasets, and the inherent complexity of skin lesions that complicate accurate classification.While the research provides valuable insights into the use of various deep convolutional neural networks, it identifies gaps in the current methodologies, particularly the underutilization of advanced architectures like EfficientNet for skin cancer classification. Additionally, it emphasizes the need for further exploration into addressing data limitations and enhancing model robustness to improve diagnostic accuracy(Sandeep Kumar et al., 2023).

The thesis titled "An Efficient Deep Learning-Based Skin Cancer Classifier for an Imbalanced Dataset" focuses on developing a robust classifier for skin cancer detection using deep learning techniques. The authors aim to address the challenges posed by imbalanced datasets, which are common in medical imaging, particularly for skin lesions. The study identifies limitations such as potential overfitting and the reliance on the quality of input images, which can vary significantly in practice. Furthermore, it highlights gaps in the research, including the need for broader exploration of transfer learning methods and real-world validation to ensure that the proposed solutions are effective in clinical settings(Alam et al., 2022).

The thesis titled "A hybrid CNN architecture for skin lesion classification using deep learning" presents a novel convolutional neural network (CNN) architecture that combines DenseNet and residual networks to enhance the classification of skin lesions. The authors aim to address common challenges in automated skin lesion detection, such as image artifacts and irregularities.While the study demonstrates promising results, it acknowledges limitations related to dataset diversity and potential overfitting. Furthermore, it highlights gaps in the research, particularly concerning the need for real-world validation and a deeper understanding of how contextual data can further improve classification accuracy(Jasil & Ulagamuthalvi, 2023).

The research identifies limitations such as reliance on high-quality images and the potential difficulty in differentiating between diseases with similar symptoms. Additionally, it highlights gaps in the need for practical validation of the model in real-world agricultural contexts and a deeper understanding of how feature extraction impacts classification outcomes (*HOG Convolutional Neural Networks and Histogram-oriented Gradients.Pdf*, n.d.).

The thesis titled "Superlative Feature Selection Based Image Classification Using Deep Learning in Medical Imaging" investigates the application of deep learning techniques for classifying medical images through enhanced feature selection methods. The authors aim to improve classifier performance by utilizing a pre-trained EfficientNetB0 model for feature extraction. The study identifies limitations such as challenges related to dataset diversity and potential overfitting, which can hinder the model's applicability in varied clinical scenarios. Additionally, it highlights gaps in the research, particularly regarding the need for real-world validation and a deeper understanding of how feature selection impacts model interpretability and performance across different medical imaging types(Humayun et al., 2022).

The research identifies limitations such as potential biases due to dataset variability and challenges with generalization, which can affect the robustness of the findings. Additionally, it highlights gaps in understanding how the model makes decisions and suggests a need for further comparison with other advanced models, as well as real-world validation to confirm its applicability in clinical practice (Sunyoto et al., 2022).

2.12. Summary of Research Gaps

In summary most of the related works have been well investigated on the CNN’s architecture feature selection parts. However, the CNN feature selection method usually is effective to capture common spatial contexts from images. Therefore, integrating other unique feature selection methods can provide the CNN model new contexts in addition the spatial information, which improved the model classification. Moreover, CNN backbone recognition performance may be enhanced by moving pre-trained weight from another domain. Finally, jointly integrating feature selection and transfer learning in CNN can further improve disease classification.

Table 1. 1 Summary of related works

Title name	Author	Objective (thesis Contribution)	Gaps
Skin cancer classification using EfficientNet architecture	(K. Ali et al., 2022)	The study focuses on classifying three types of skin cancer (BCC, SCC, melanoma) using the EfficientNet architecture. It preprocesses images, trains models from EfficientNet-B0 to EfficientNet-B7, and evaluates their performance metrics.	Lack of exploration into more types of skin cancer and real-world applicability.
Classification of Skin Cancer Images using Convolutional Neural Networks	(Sandeep Kumar et al., 2023)	The research aims to develop a Deep Convolutional Neural Network (DCNN) model for automatically classifying skin cancer types into melanoma and non-melanoma using dermoscopic images, focusing on high accuracy in classification.	• Limited application of EfficientNet models in skin lesion classification despite their potential. • Need for more robust solutions addressing data imbalance and domain adaptation. • Exploration of multimodal approaches and lightweight models for better efficiency.

An Efficient Deep Learning-Based Skin Cancer Classifier for an Imbalanced Dataset	(Alam et al., 2022)	• The study may be limited by the size and diversity of the dataset used, which can affect generalizability. • Potential overfitting due to class imbalance despite augmentation techniques. • Dependency on the quality of input images which can vary in clinical settings.	• Insufficient exploration of transfer learning methods beyond those implemented. • Need for more comprehensive evaluation metrics to assess model performance across different demographics. • Lack of real-world validation in clinical environments to confirm efficacy and reliability.
Convolutional Neural Networks and Histogram-Oriented Gradients: A Hybrid Approach for Automatic Mango Disease Detection and Classification	(HOG Convolutional Neural Networks and Histogram-oriented Gradients.Pdf, n.d.)	• The model's performance may be affected by the quality and variability of the input images. • Challenges in distinguishing between similar diseases due to overlapping features. • Dependence on extensive labeled datasets for training, which can be resource-intensive to obtain.	• Insufficient exploration of real-world applicability in diverse agricultural settings. • Limited investigation into the interpretability of the model's decisions. • Need for further research on optimizing feature extraction techniques for enhanced classification accuracy.

A hybrid CNN architecture for skin lesion classification using deep learning	(Jasil & Ulagamuthalvi, 2023)	Challenges with artifacts in images such as hair and irregular lesion shapes can complicate classification. • The model's performance may be limited by the diversity and size of the dataset used for training. • Potential overfitting issues due to reliance on specific datasets without extensive validation across varied populations.	• Insufficient exploration of real-world applicability and validation in clinical settings. • Need for further investigation into the integration of additional contextual data for improved accuracy. • Limited analysis of the model's interpretability and how it handles edge cases in skin lesion classification
• Superlative Feature Selection Based Image Classification Using Deep Learning in Medical Imaging	• (Humayun et al., 2022)	• - Dependence on the quality and diversity of the dataset, which may limit the model's generalizability. • Potential overfitting due to imbalanced datasets and reliance on specific features. • The effectiveness of the	• Insufficient exploration of real-world clinical applications and validation. • Limited discussion on the interpretability of the model's decisions and feature importance. • Need for further research on optimizing feature selection

		proposed methods may vary across different medical imaging modalities.	techniques for broader medical imaging contexts.
The Performance Evaluation of Transfer Learning VGG16 Algorithm on Various Chest X-ray Imaging Datasets for COVID-19 Classification	(Sunyoto et al., 2022)	• The model's performance may be affected by variations in image quality across different datasets. • Challenges in generalizing results due to dataset bias and limited diversity in training samples. • Potential overfitting when using small datasets for training.	• Insufficient exploration of the model's interpretability and decision-making process. • Limited comparison with other state-of-the-art models beyond VGG16. • Need for real-world validation to assess the model's effectiveness in clinical settings.

CHAPTER THREE

Research Methodology

3.1. Chapter Overview

This chapter provides the methods, approaches, and strategies utilized to improve the quality of dermatology images, which is the primary goal of the thesis work. The datasets utilized, data augmentation strategies, preprocessing procedures, development tools, baseline work, and assessment methodologies are all covered.

Currently, in order to determine whether a patient has a skin condition or not, a dermatologist must perform a single test to check for skin cancer. Dermatologists can handle a variety of cases far more quickly because to this platform (Hasan et al., 2019).

3.2. Research Approach and Procedure of the Study

This study fits under the genre of quantitative research, which may yield findings through experimentation, given the nature of the topic at hand and the research objectives of the thesis.

In light of this, the techniques are research design techniques that include data gathering and analysis, data quantification, relationship construction support a theory (Sam, 2012).

Several various methods have been used to achieve the research work's objective. To achieve the desired outcome of the planned work, those are a series of steps or actions. Identification of the research area, evaluation of prior work and knowledge by different researchers, gap analysis, formulation of the research question, tool and methodology selection, dataset preparation, and, at the end, designing and implementing the optimal solution are the primary processes or procedures utilized in the operation of this thesis. The following figure depicts a great view of the actions.

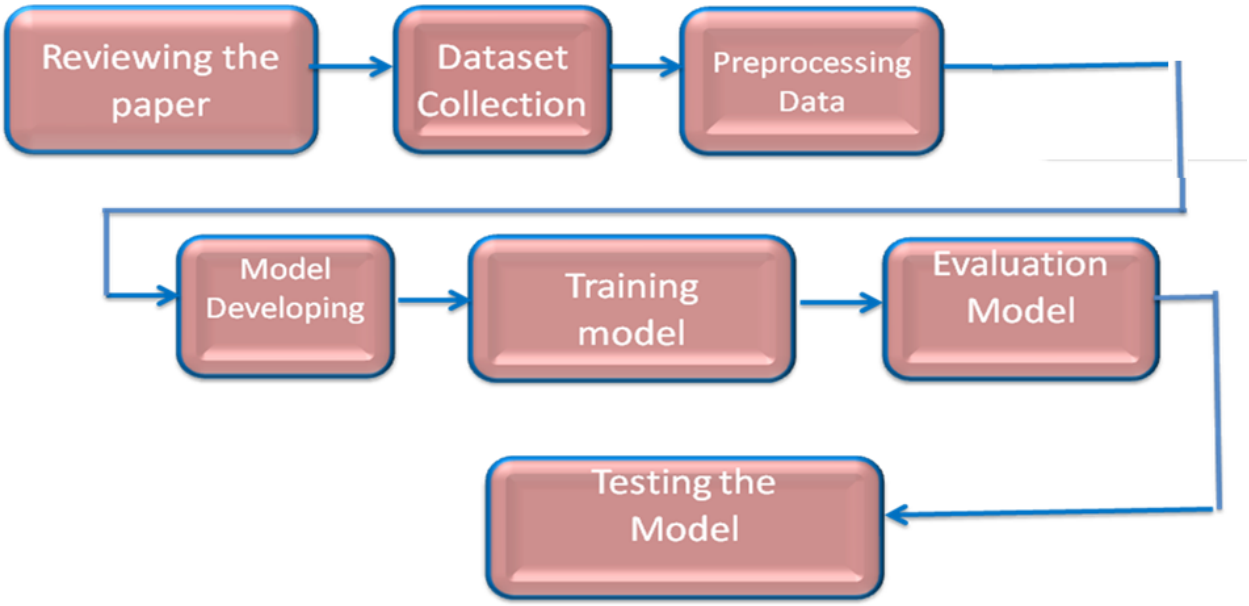


Figure 3. 1 Research methodology

Literature review: To learn more about the most recent advancements in the classification system for skin cancer, we reviewed relevant journals, academic thesiss, websites, and other sources.

System design: The workflow and system architecture, as well as the data flow from the skin cancer classification system to the database, will be designed by us.

Data Collection: For this study, we make an effort to exploit the Kaggle open-source HAM10000 dataset.

Model development: The deep learning model for skin cancer categorization will be created on the GPU server. The model will be trained on a dataset of classified cases of skin cancer, and its performance will be assessed using metrics like accuracy

System evaluation: By assessing the system's accuracy and usability, we will assess how well it performs.

3.3. Research design

Research design, or the general plan or strategy for carrying out a research endeavor, is an important part of the research technique. In order to address the study issues, the procedures and methods for data collection and analysis are decided upon in this part. There are the following elements of the research design:

Research question: It is imperative that the research question posed in the first chapter be specific and focused.

Research approach: The general plan for carrying out the investigation is referred to as the research approach. In this instance, a quantitative research methodology will be suitable, since the study's goal is to gauge the system's correctness and efficiency.

Data collection methods: The specific procedures utilized to acquire data are referred to as data collecting methods. In this instance, images of skin cancer will be used to collect data.

Data analysis methods: The precise statistical or qualitative procedures used to examine the data that was gathered are referred to as data analysis methods. In this instance, the metrics used for data analysis will include accuracy, which will be used to gauge the system's effectiveness and performance.

3.3.1. Dataset

Using a dataset of 10,015 images, we will assess the suggested CNN model's performance in this study. From this 10,015 only we use about 3145 images by dividing in to three of 90% for training, 10% for validation and from each class we select 100 images for each class to testing. The images will be collected from ISIC, the International Skin Imaging Collaboration HAM10000 database (Botunac 2020), because of the availability and quality for evaluation purpose. Since this study will follow Deep Learning approach to solve a classification of skin cancer, data is an essential part of the study. In medical industry, particularly in Ethiopia collecting and annotating image dataset is very difficult and worth much time and costy, and hence, the use of public dataset is important.

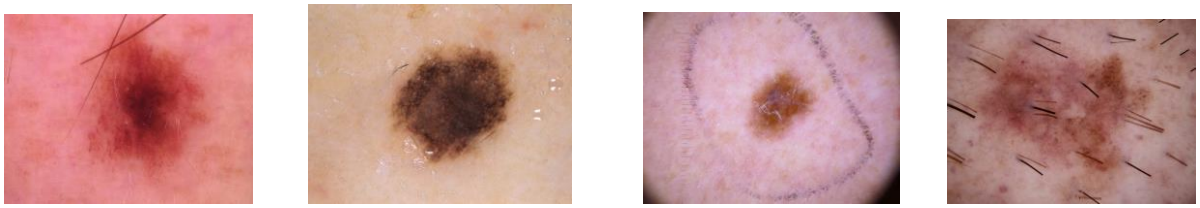


Figure 3. 2 HAM10000 dataset sample examples

3.3.2. Data Augmentation

A training set can be made larger by employing image data augmenting to include significantly modified copies of the latest data or newly generated synthetic data from the existing data. It acts as a regularizer and lessens over-fitting in machine learning models during training. When fit models are trained on additional data, they become more proficient at applying what they have learnt to new images. Large datasets yield better results from DL models. We all know that jittering or data augmentation is a popular way to increase the size of our databases. To avoid overfitting with insufficient data, data augmentation can increase the dataset's size by up to ten times or more. The method assists in creating more robust, straightforward models that have improved generalization. The simplest methods for doing data augmentation are to geometrically alter or add noise to already-existing data.

3.4. Development Tools

3.4.1. Hardware Tools

In the table below, you'll find a list of the hardware tools that were used throughout the study.

Table 3. 1 Hardware tools

Number	The Device Name	Usage for the research
1	GPU	To train the model
2	Hard disk	To store the datasets and models
3	RAM	To improve hardware performance

3.4.2. Software Tools

The research is implemented by coding using a variety of software and coding tools, some of which are listed below with descriptions.

Anaconda[5]: is a Python and R combination that makes package management and distribution easier. With just one installation, Anaconda is renowned for providing a broad range of machine learning and data analysis capabilities. The Anaconda package manager, Conda, can be used to install libraries. Additionally, it features a useful interface with pip that makes it possible to install libraries that cannot be accessed using Anaconda's package manager. It works with Linux, macOS, and Windows and is not dependant on any particular platform.

Tensor flow: The public can use a software program that uses machine learning techniques, particularly neural networks. It is a free, open-source, end-to-end platform for machine learning. It can also be used on desktop PCs or in the cloud.

Jupyter Notebook[6]: An open-source online application that combines computer output, documentation, video, and software code into a single document. It is used to change code and view results, and it is integrated with the Anaconda environment.

PyTorch[7]: A machine learning framework available as open source that expedites the process from research prototyping to production deployment. Its foundation is the Torch library, which is employed in computer vision and natural language processing applications. PyTorch is nearly as easy to learn as the Python programming language because its syntax is similar.

OpenCV[8]: is a programming function library with a focus on real-time computer vision. It was initially created by Intel and subsequently sponsored by Willow Garage. The Apache 2 License allows for the free and open usage of the cross-platform library.

Microsoft office package: It is used to prepare the study document and create slides for presentation, and Visio is used to create the proposed system's diagrams and flowcharts.

Medley desktop: It is well known for its reference manager, which is used to organize and exchange research thesiss, as well as create bibliographies for scholarly publications and references to related works.

3.4.3. Model evaluation techniques

The dataset was eligible for holdout validation since it included a sufficient number of samples for both testing and training. After the model training was finished, the CNN, HOG, and hybrid models' performances were assessed using the testing dataset. The performance was assessed using a wide range of widely used metrics, including as sensitivity (recall), accuracy, precision,

and F1 score. The general correctness of the model's predictions or the classification accuracy of the validation (training) data are both quantified by accuracy. The number of true positives, true negatives, false positives, and false negatives was determined using a confusion matrix, which assisted in assessing the efficacy of the suggested model (*HOG Convolutional Neural Networks and Histogram-oriented Gradients.Pdf*, n.d.).

Accuracy: - Accuracy is one measure used to assess how well a model performs on a collection of data. Divide the total number of projections by the total number of precise forecasts to find it.

(TP + TN)/(TP + FP + TN + FN) is the accuracy.

Precision: - The precision of a model's positive predictions indicates how accurate it is. According to its definition, it is the ratio of actual positive outcomes to all positive expectations. These measures, which show the model's effectiveness in every class, are calculated based on classification outcomes.

Reliability = TP −(TP + FP)

Recall: - A measure of a model's accuracy in making comprehensive predictions is its recall. It can be computed by dividing the total number of real positives by the percentage of true positives.

Remember = TP − (TP + FN)

F-1 Score: - The F1 score is a model's performance metric that combines precision and recall. One is the highest F1 score, and zero is the lowest. The harmonic mean of recall and precision is how it is defined. Both accuracy and recall have an equal effect on the F1 score.
F1 = 2 ∗ (recall ∗ precision) ∗ (recall + precision)

With a focus on categorizing skin cancer, the model is trained to predict whether or not a picture contains characteristics in the skin tissue that are suggestive of a disease.

Confusion matrix: A visual aid for understanding a classification model's performance is a confusion matrix. The expected and actual class labels are shown in a tabular format.

CHAPTER FOUR

Proposed Model

4.1. Chapter Overview

The suggested model for creating a human image from a particular stance is presented in this chapter. Better presentation and content preservation of the original image should be features of the created image. As a result, it mainly presents the model architecture, training plans, and loss function optimization techniques together with the different preprocessing methodologies to produce the image from posture.

4.2. The Proposed Model

The aim of this work was to develop a predictive model for the categorization of skin cancer using picture data by utilizing deep learning techniques. The model predicts whether skin samples are indicative of different types of skin cancer, such as actinic keratoses and intraepithelial carcinoma (AKIEC), melanoma (MEL), vascular lesions (VASC), basal cell carcinoma (BCC), and benign lesions of the keratosis (BKL), using a dataset of images of skin lesions. The proposed model explains the initial stage of obtaining preprocessed image data.

.

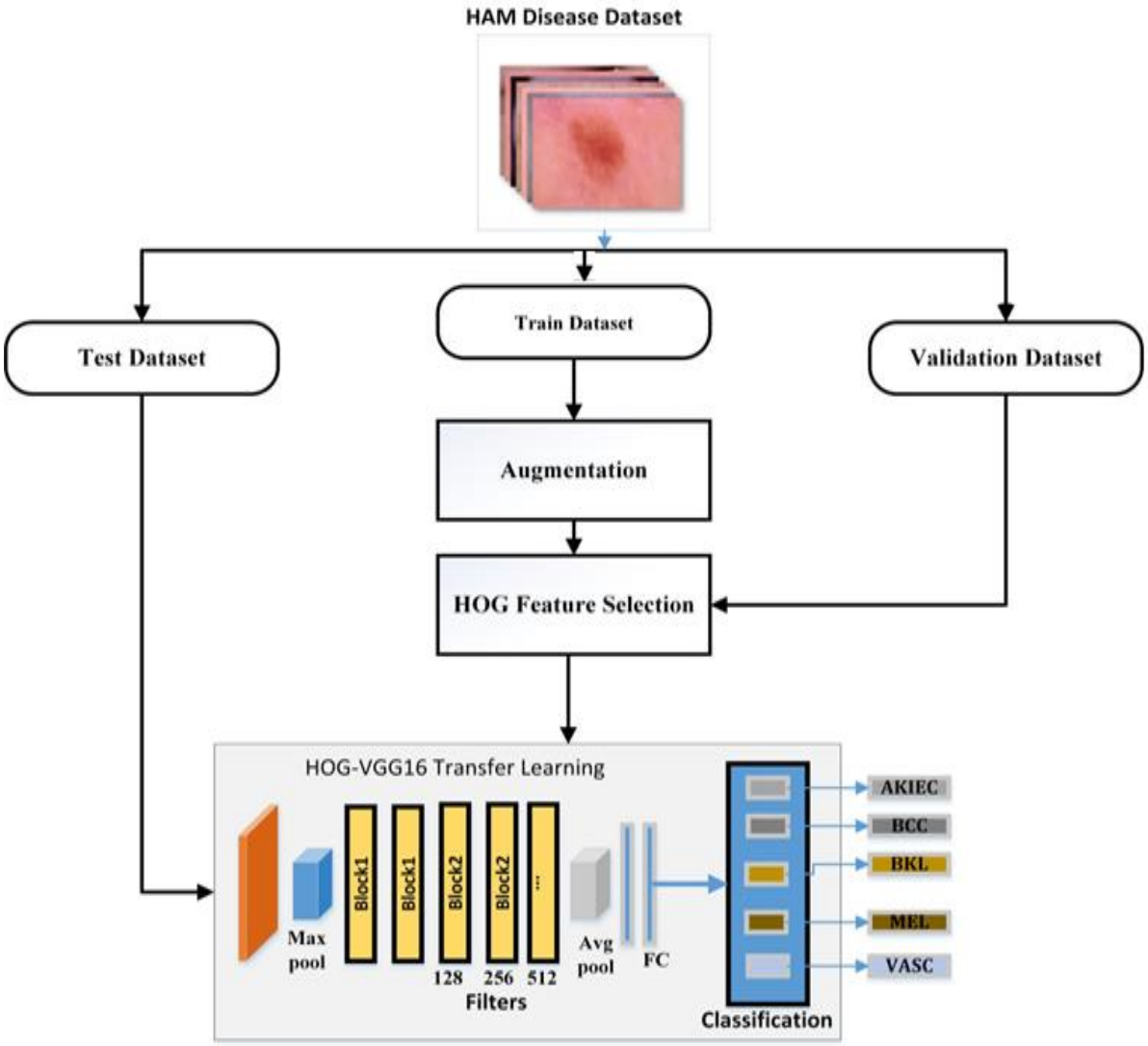


Figure 4. 1 The Proposed Model

4.3. Dataset processing

Data processing involves tasks such as noise reduction, image enhancement, segmentation of skin lesions from unwanted background, and normalization to ensure uniformity across images.

After preprocessing the dataset, techniques such as feature extraction, lesion segmentation, and image augmentation can be applied to enhance the dataset for training the skin cancer classification model.

Having balanced the dataset and preprocessed it, the VGG16 model was utilized as the foundation for constructing a predictive model for skin cancer classification. A variety of

performance measures were used to evaluate this model's performance in order to identify the most effective technique for categorizing skin cancer cases.

```
def preproccessing():
    image_dir ="/HAM1000" #Generated image directory as trainingset
    imgFiles_train = [
        join(image_dir, fn)  # Create full paths to images
        for fn in os.listdir(image_dir)  # For each item in the image folder
        if isfile(join(image_dir, fn))  # If the item is indeed a file
           and fn.lower().endswith(('.png', '.jpg'))
        ]
```

Dataset collection: To train a machine learning model—like a convolutional neural network (CNN)—for the classification of skin cancer, data must be gathered and arranged. The process of gathering data aims to cover a broad range of skin disorders in addition to normal skin. Thus, to achieve our goal, we make use of publicly accessible web databases that include images of skin lesions. The Skin Cancer Images dataset on Kaggle includes 10,015 pictures of skin lesions.

Dataset organization: The study's picture datasets are classified and divided into training, testing, and validation sets. While the training set trains the model to recognize patterns, the validation set evaluates the model's capacity to generalize to previously unseen data. Since epoch counts range from 1 to 40, selecting the appropriate number is crucial to preventing either underfitting or overfitting. Strategies like adding more data, growing the network, and extending epochs can be used to increase model accuracy. Additionally, the thesis gives results on the use of mammography imaging for skin cancer categorization utilizing a locally collected dataset retrieved from Kaggle. The categories for the 10015 mammography skin cancer images are: actinic keratoses and intraepithelial carcinoma (AKIEC), basal cell carcinoma (BCC), benign lesions of the keratosis (BKL), melanoma (MEL), and vascular lesions (VASC). The dataset was split into three parts: training, testing, and validation, in order to evaluate the model's performance in classification.

Intraepithelial carcinoma and actinic keratoses (AKIEC): These are precancerous lesions that can develop on the skin, often caused by sun exposure. They are flat, scaly patches that may be red, pink, or brown. While most AKIECs do not develop into cancer, some can progress to squamous cell carcinoma.

Basal Cell Carcinoma (BCC): This type of skin cancer is the most common. It usually appears as a raised, pearly, waxy mass with often visible blood vessels. Although BCCs have the potential to spread locally, they rarely do so.

Benign Lesions of the Keratosis (BKL): These are non-cancerous skin growths that can resemble cancerous lesions. Examples include seborrheic keratoses and actinic keratoses.

Melanoma: The most dangerous kind of skin cancer is melanoma, which can spread to other body areas. A mole that changes in size, form, or color is frequently the first sign of melanoma. They can also be unexpectedly developing new moles.

Vascular Lesions: These are skin conditions caused by abnormal blood vessels. Examples include hemangiomas, which are benign growths of blood vessels, and telangiectasias, which are dilated blood vessels that appear as red or purple lines.

Dataset Setting: In the course of working with the HAM10000 ("Human against Machine with 10000 training images") dataset(Tschandl et al., 2018), adjustments were made to address class imbalance. Specifically, a subset of the dataset was extracted by removing two classes, one with the highest number of images (6707) and the other with the lowest number of images (162). This was done to make sure that the data was distributed more fairly among the remaining classes. A random sample of 100 photos from the original dataset was reserved for testing in order to evaluate the efficacy of the suggested method. The remaining photos were also split into two groups: 10% of the data was used for validation and 90% of the data was used for training.

This partitioning strategy enables us to train our model on a substantial portion of the data while also having a separate set for validation and fine-tuning. In an effort to expand the training size, the training data was augmented to reach 1000 samples. We have removed images that are below the minimum width and height of 112x112 to ensure consistency.

Table 4. 1 Train, test and validate Original ratio on the dataset

Original Dataset

Disease Class	Original
Actinic keratoses and intraepithelial carcinoma[AKIEC] 0	327
Basal Cell Carcinoma [BCC]	414
Benin Lesions of the Keratosis [BKL] 2	1099
DermatoFibroma [DF] 3	165
Melanoma [MEL] 4	1113
Melanocytic Nevi [NV] 5	6705
Vascular Lesions [VASC] 6	192

Table 4. 2 Train, test and validate split ratio on the dataset

Dataset after removing imbalance classes

Disease Class	Original	90% Train	10 % Val	Test	Augment
Actinic keratoses and intraepithelial carcinoma(AKIEC)	327	227	23	100	1000
Basal cell carcinoma (BCC)	414	314	31	100	1000
Benign lesions of the keratosis (BKL)	1099	999	100	100	1099
Melanoma (MEL)	1113	1013	101	100	1113
Vascular lesions (VASC)	192	92	9	100	1000

4.4. Data processing Techniques

Normalization: Normalization is one technique used during the data preparation phase. Consequently, the values of the numeric columns in the data set are changed to utilize the common scale of [-1, 1] when the features of the data have various ranges. Additionally, the following calculations are made using the image that is provided:

.

```
    ##################### transforms.Normalize([0.485, 0.456, 0.406], [0.229, 0.224, 0.225])#################
    train_loader = hog(torch.utils.data.DataLoader(
        DataLoader.ImageList(
            imgFiles_train, transform=transforms.Compose([transforms.ToTensor(), transforms.Normalize((0.5, 0.5,
                                                                                                       0.5), (0.5, 0.5,
                                                                                                              0.5))])),
        batch_size=batch_size, num_workers=0, sampler=train_loader, drop_last=True, pin_memory=True))

    valid_loader = hog(torch.utils.data.DataLoader(
        DataLoader.ImageList(imgFiles_train, transform=transforms.Compose([transforms.ToTensor(), transforms.Normalize((0.5, 0.5,
                                                                                                       0.5), (0.5, 0.5,
                                                                                                              0.5))])),
                                                                                                              batch_size=batch_size,
num_workers=0,
                                                                                                              sampler=valid_loader,
drop_last=True,
                                                                                                              pin_memory=True))
```

Figure 4. 2*The cod for normalizing the image*

Resize Image- The dataset's sets of sizes differ; therefore they must be changed to fit the input size of the network. The reason for this resizing is to reduce computing complexity and to crop the region of interest in order to focus on it. They are scaled to 112x112 in this work, which is in line with the work of (Chen et al., 2019), and this resizing is important for reducing a computational complexity and also to focus on the region of interest by cropping the region we are interested on.

Data Augmentation- For the deep learning process (deep network) to function well, large datasets are typically needed. Therefore, data augmentation is a crucial tool for improving performance when there is a lack of available data. Augmentation is the process of artificially producing training images by several preprocessing methods, including random rotation, flipping, shifting, and other methods. It is an excellent method for increasing the size of your dataset to get better outcomes.

```
In [2]: import Augmentor

In [ ]: #folder
        p = Augmentor.Pipeline("C:/Users/gate/Desktop/EDOSA/HAAM/0/")
        # p = Augmentor.Pipeline("C:/Users/gate/Desktop/EDOSA/HAAM/1/")
        # p = Augmentor.Pipeline("C:/Users/gate/Desktop/EDOSA/HAAM/2/")
        # p = Augmentor.Pipeline("C:/Users/gate/Desktop/EDOSA/HAAM/3/")
        # p = Augmentor.Pipeline("C:/Users/gate/Desktop/EDOSA/HAAM/4/")

        p = Augmentor.Pipeline("./HAM1000/")

In [3]: p.random_distortion(probability=1, grid_width=4, grid_height=4, magnitude=8)
        p.flip_left_right(probability=0.5)
        p.flip_top_bottom(probability=0.5)
        p.rotate90(probability=0.5)
        p.rotate270(probability=1)
        p.crop_random(probability=0.8, percentage_area=0.8)
        p.zoom(probability=0.8, min_factor=1.2, max_factor=1.6)
        p.resize(probability=1.0, width=128, height=128)
```

Figure 4. 3 *The process of image augmenting*

4.5. HOG for feature selection and VGG16 for classification

The performance of the suggested method based on VGG16 classification and HOG feature selection is shown in the following experiment. First the pre-processed images are given to HOG feature selector, then the feature generated or extracted by HOG module is given to VGG16 model for classification.

```
: import matplotlib.pyplot as plt
  from skimage import io
  from skimage.feature import hog
  from skimage import color

  # Load an image
  image = io.imread('images/ISIC_0024545.jpg')

  # Convert the image to grayscale
  image_gray = color.rgb2gray(image)

  # Calculate the HOG features
  hog_features, hog_image = hog(image_gray, visualize=True)

  # Plot both the original and HOG images
  plt.figure(figsize=(12, 6))

  # Plot the original image
  plt.subplot(1, 2, 1)
  plt.imshow(image)
  plt.title('Original Image')
  plt.axis('off')

  # Plot the HOG image
  plt.subplot(1, 2, 2)
  plt.imshow(hog_image)
  plt.title('HOG Image')
  plt.axis('off')

  plt.show()
```

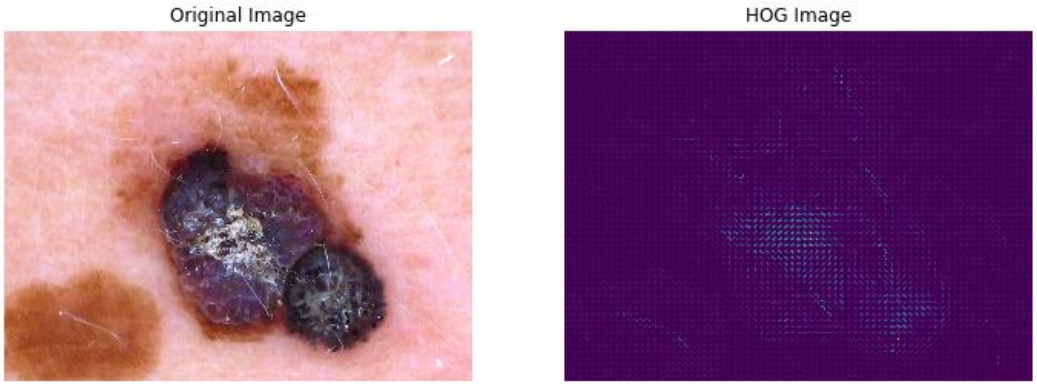


Figure 4. 4 T*he HOG image reader*

HOG Feature

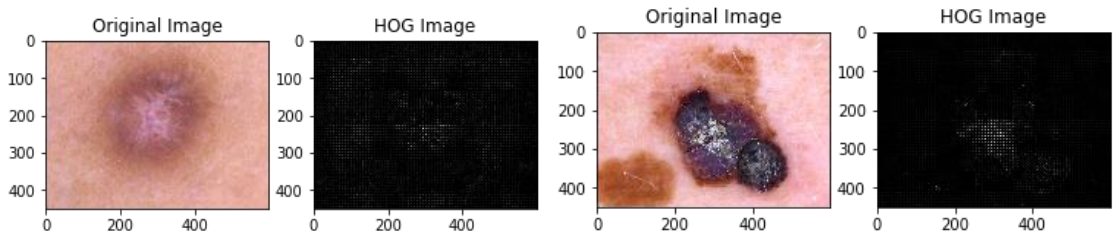


Figure: 4.2.HOG Feature selector

A CNN model pulls pertinent characteristics from the input data using a range of feature selection methods. Among the feature selection methods CNNs employ are:

Convolutional Filters: The initial layer of a CNN, the convolutional layer, filters incoming data. These filters, which are acquired through training, are utilized to extract characteristics from the input data. The filters are designed to go across the input data and convolve each pixel with its neighbors in order to extract local features.

Pooling: The output is run via a pooling layer subsequent to the convolutional layer. By utilizing the maximum or average value of the data inside a window, the pooling layer minimizes the spatial dimensions of the data. This contributes to lowering the model's required amount of computations and parameters.

Flattening: The output is flattened into a one-dimensional vector after the pooling layer. This flattening aids in lowering the dimensionality of the data and enables a more linear processing of the features by the model.

Dropout is a regularization method that CNNs employ to avoid overfitting. During training, it arbitrarily sets a portion of the neurons to zero, so generating an ensemble of distinct sub-networks. By doing this, the model is kept from depending too much on any one feature.

Activation Functions: CNNs use activation functions like as Tanh, Sigmoid, and ReLU (Rectified Linear Unit) to introduce non-linearity into the model. These activation functions help to amplify the signal and provide more informative features.

CNNs use a technique called batch normalization to standardize the input data for each layer.

Weight Pruning: is a method for cutting down on a CNN's parameter count. It entails eliminating weights from the model that are superfluous or redundant, which helps to increase model efficiency and lower the likelihood of overfitting.

With the aid of the aforementioned feature selection approaches, CNNs are able to extract pertinent features from the input data and produce a reliable representation of the data that is suitable for use in tasks like segmentation, object identification, and picture classification.

Classification: There are numerous methods for classifying skin cancer with CNNs. The particular application, the quantity and caliber of the dataset, the required degree of precision, and the feature extraction photos all influence the technique selection. Thus, we suggest feature level classification as a way to extract features from the processed image and obtain the desired output. Using this method, the CNN model is trained to identify and categorize characteristics that are extracted from the input image. For every image, the model produces a classification score as well as a set of feature maps.

CHAPTER FIVE

Results, Evaluation, and Discussions

5.1. Chapter Overview

This chapter explores the experimental findings from the research, offering a thorough rundown of model performance, assessment criteria, and preprocessing methods for image data. A thorough comparison with previous studies is also given to demonstrate how this study addresses the research topic, closes any gaps in the literature, and achieves the specific objectives outlined.

5.2. Data Augmentation results

A popular machine learning technique for expanding the amount and variety of a training dataset is data augmentation. Artificially increasing the dataset helps the model become more resilient and enhances its capacity to generalize to new, unseen data. Data augmentation can be especially helpful when classifying skin cancer because the dataset's quantity and variety may be restricted. In this section, we will look at the results of applying data augmentation to a dataset for skin cancer classification. The same experimental design and dataset as in the previous part will be used, but with an extra augmentation phase.

5.3. VGG16 for classification and HOG for feature selection

HOG (Histogram of Oriented Gradients) and VGG-16 are two powerful techniques that can be combined effectively for various computer vision tasks, particularly image classification.

HOG for Feature Selection: Local Feature Extraction, HOG extracts local features based on the distribution of edge orientations in an image and Invariant to Geometric Transformations is HOG features are relatively invariant to changes in scale, rotation, and illumination and Dimensionality Reduction HOG can reduce the dimensionality of the feature space compared to raw pixel values.

VGG-16 for Classification is a deep learning architecture that has been pre-trained on a large dataset like ImageNet and it has high-level feature extractor which has capable of extracting high-level semantic features from images.

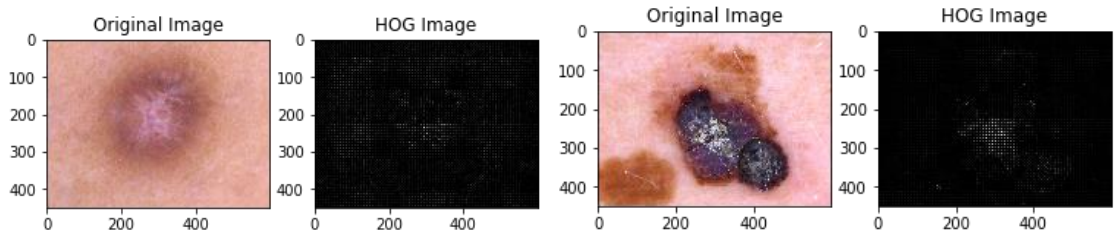


Figure 5. 1 HOG Feature selection

5.4. The VGG 16 classification result

Known for its simplicity and depth, VGG-16 is a well-liked convolutional neural network (CNN) architecture that frequently produces outstanding results when used for classification tasks. As shown from Figure 5.2 the classification accuracy is 90.6% which is good enough in indicating skin cancer in test images.

VGG16 Classification Report:

	precision	recall	f1-score	support
AKIEC	0.887	1.000	0.940	102
BCC	1.000	0.898	0.946	118
BKL	0.912	0.989	0.949	94
MEL	0.802	0.880	0.839	83
VASC	0.919	0.767	0.836	103
accuracy			0.906	500
macro avg	0.904	0.907	0.902	500
weighted avg	0.911	0.906	0.905	500

Figure 5. 2 The VGG 16 classification result

5.5. The VGG 16 Confusion matrix result

From the confusion matrix given in Figure 5.3, the confusion matrix reveals that the VGG16 model misclassified a total of 47 out of 500 test samples, resulting in an error rate of 9.4%. i.e. (error=47/500*100=9.4%).

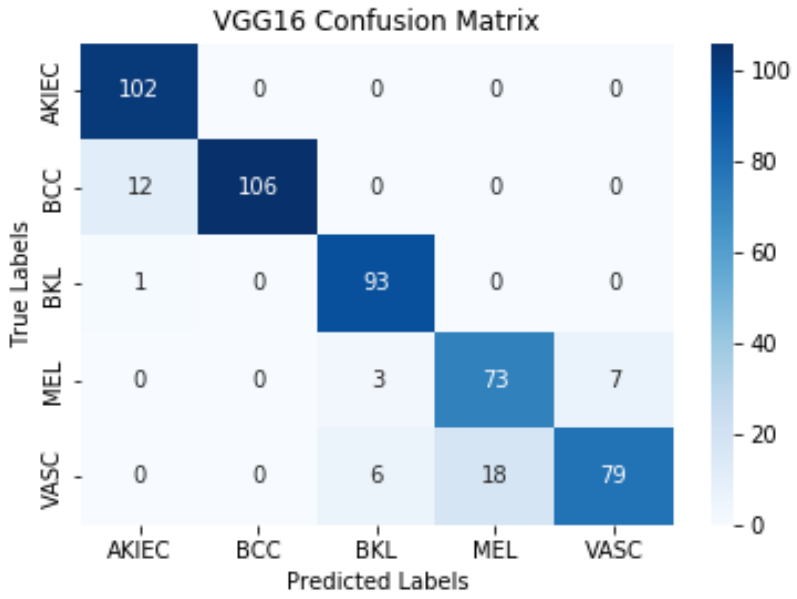


Figure 5. 3 The VGG 16 confusion matrix

5.6. The VGG 16 training curve result

The accuracy and loss curves for training and validation over 40 epochs are shown in Figure 5.4. Throughout the training phase, the baseline VGG16 model continuously showed increased accuracy and decreased loss. This shows that the model can keep learning from the dataset all the way up to the 40th epoch. But after that, as Figure 5.5 shows, the model converged to a stable state.

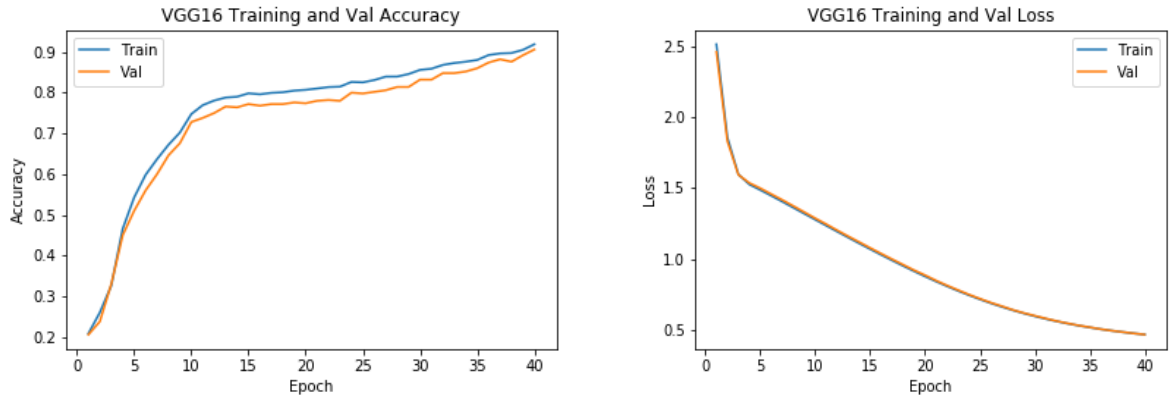


Figure 5. 4 The VGG 16 training curve result

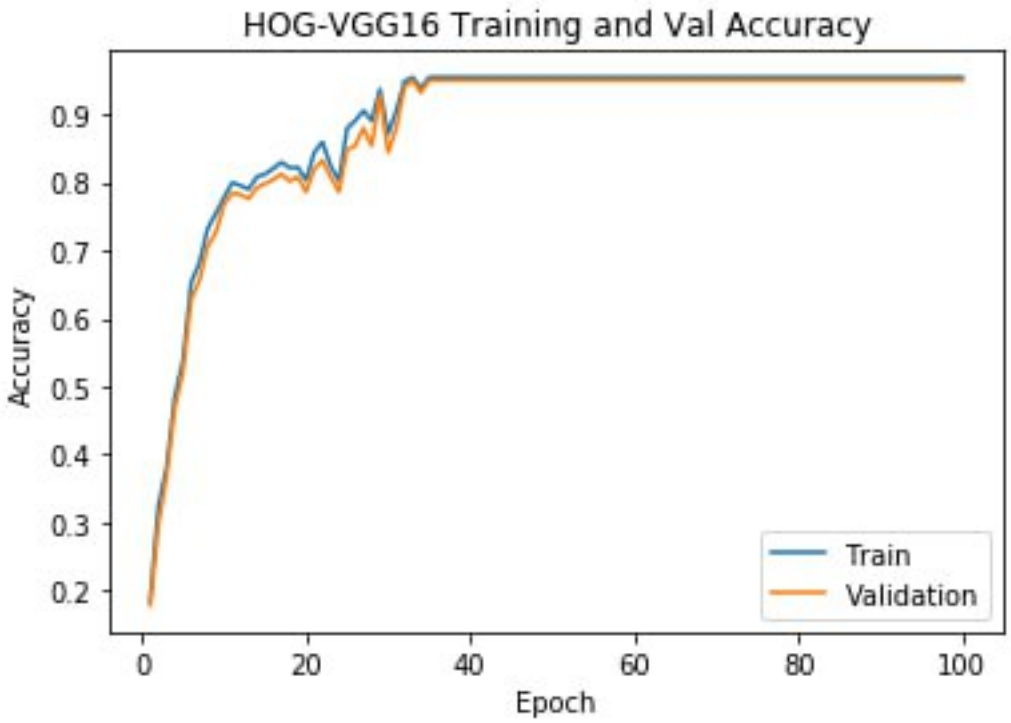


Figure 5. 5 The VGG16 training curve showing constant on 40 epoch

5.7. The VGG 16 + HOG classification result

We evaluated the performance of VGG16 with HOG feature selection method. As shown in Figure 5.5, the performance of VGG16 with HOG improved by (90.6-95.0=4.4) +4.4% which indicates using feature selection can boost classification performance.

HOG-VGG16 classification Report:

	precision	recall	f1-score	support
AKIEC	0.887	1.000	0.940	102
BCC	1.000	0.898	0.946	118
BKL	0.912	0.989	0.949	94
MEL	0.964	0.964	0.964	83
VASC	1.000	0.913	0.954	103
accuracy			0.950	500
macro avg	0.953	0.953	0.951	500
weighted avg	0.954	0.950	0.950	500

Figure 5. 6 The VGG 16 and HOG classification result

5.8. VGG 16 + HOG confusion matrix result

We are utilizing the advantages of both methods when we combine Histogram of Oriented Gradients (HOG) with VGG-16, a deep convolutional neural network. High-level semantic characteristics in images are well captured by the VGG 16, and local edge and shape information is effectively captured by the HOG (Histogram of Oriented Gradient). We frequently obtain superior classification performance by concatenating the features extracted from these two approaches compared to using VGG16 without HOG. As shown in Figure 5.6 there only 25 confusion out of 500 test samples reducing the number of confusions almost by half which indicates improving skin cancer diagnosis precision.

From the confusion matrix given in Figure 5.7, the confusion matrix reveals that the VGG16 model misclassified a total of 25 out of 500 test samples, resulting in an error rate of 5%. i.e. (error=25/500*100=5%).

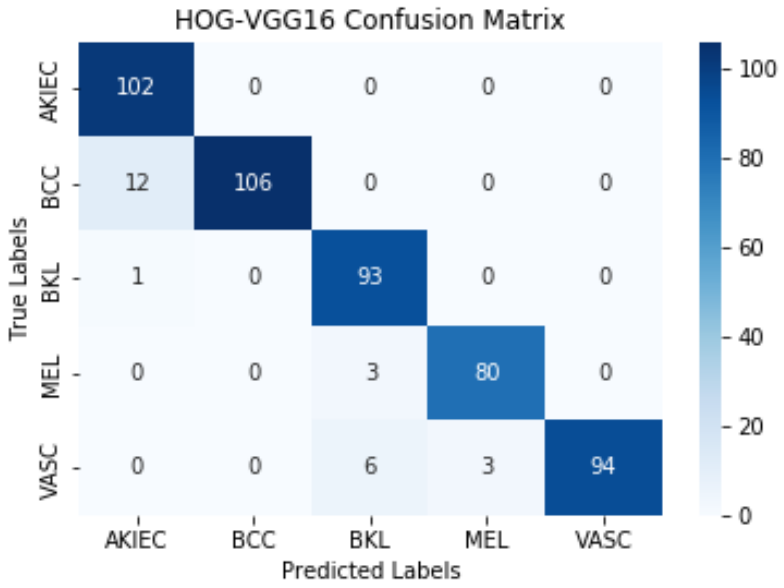


Figure 5. 7 The HOG VGG 16 confusion matrix

5.9. VGG 16 + HOG training curve result

A training curve shows the accuracy and loss function of a model as a function of time (epochs), and it provides a visual representation of the model's performance during training. When VGG16 and HOG (Histogram of Oriented Gradients) are combined, the training curve can provide insight into the way the model is learning. The model's performance on the training set is represented by the training accuracy curve. It usually starts high and, if overfitting occurs, may reduce slightly, but it should generally rise over time. Additionally, the model's performance on a different validation set is represented by the validation accuracy curve. Given that it should ideally rise in tandem with training accuracy, it can aid in the identification of overfitting.

The training and validation accuracy and loss curves across 40 epochs are shown in Figures 5.8 and 5.9. Throughout the training phase, it also continuously showed better accuracy and less loss than the baseline VGG16 model and HOG. This shows that the model can keep learning from the dataset all the way up to the 40th epoch. The model did, however, converge to a stable state beyond this point.

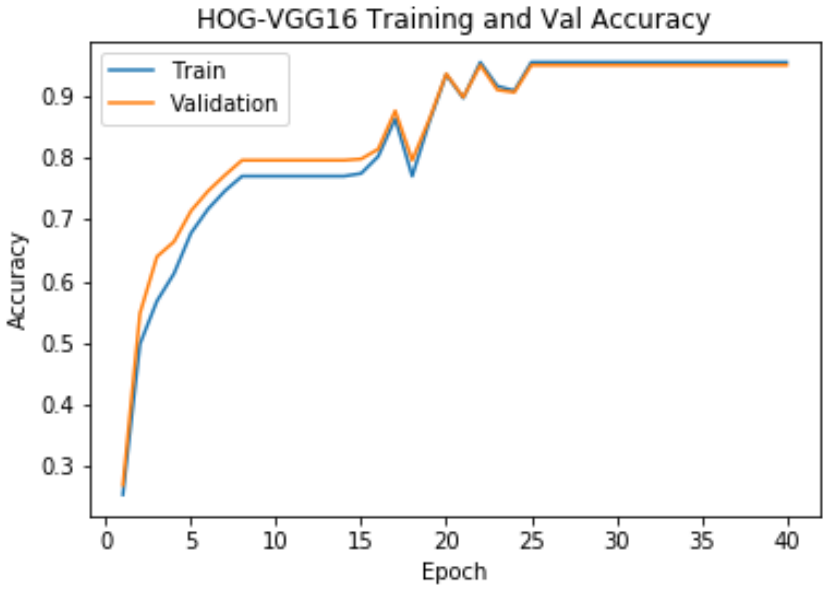


Figure 5. 8 The HOG – VGG16 training and val accuracy curve

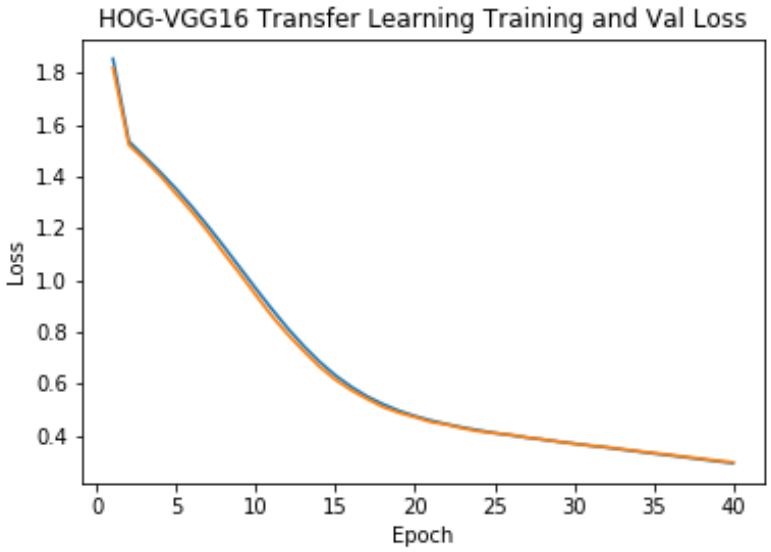


Figure 5. 9 The HOG – VGG16 training and Val Loss curve

5.10. HOG-VGG16 with transfer learning

We evaluated the performance of VGG16; HOG feature selection and Transfer learning method. As shown in Figure 5.10, the performance of VGG16+ HOG + transfer learning improved by (95.0-96 =1.0) = 1.0% which indicates using feature selection with transfer learning can boost classification performance.

HOG-VGG16-Transfer Learning Classification Report

	precision	recall	f1-score	support
AKIEC	0.944	0.990	0.967	102
BCC	0.982	0.949	0.966	118
BKL	0.912	0.989	0.949	94
MEL	0.964	0.964	0.964	83
VASC	1.000	0.913	0.954	103
accuracy			0.960	500
macro avg	0.960	0.961	0.960	500
weighted avg	0.962	0.960	0.960	500

Figure 5. 10 The classification result of HOG - VGG 16 with TL

5.11. The confusion matrix of HOG – VGG 16 with Transfer Learning

We are utilizing the advantages of both techniques when we combine the deep convolutional neural network VGG-16 with the Histogram of Oriented Gradients (HOG) and Transfer learning. The HOG (Histogram of Oriented Gradient) efficiently captures local edge and shape information and the transfer learning is maximizing the performance whereas the VGG 16 does a good job of capturing high-level semantic features in images. When we combine the features retrieved from these three methods, we often achieve better classification performance than when we use VGG16 with HOG. Figure 5.11 illustrates that out of 500 test samples, there were only 20 cases of confusion, nearly halving the total number of confusions. This may indicate an improvement in the categorization accuracy of skin cancer.

From the confusion matrix given in Figure 5.11, the confusion matrix reveals that the VGG16 model misclassified a total of 20 out of 500 test samples, resulting in an error rate of 4%. i.e. (error=20/500*100=4%).

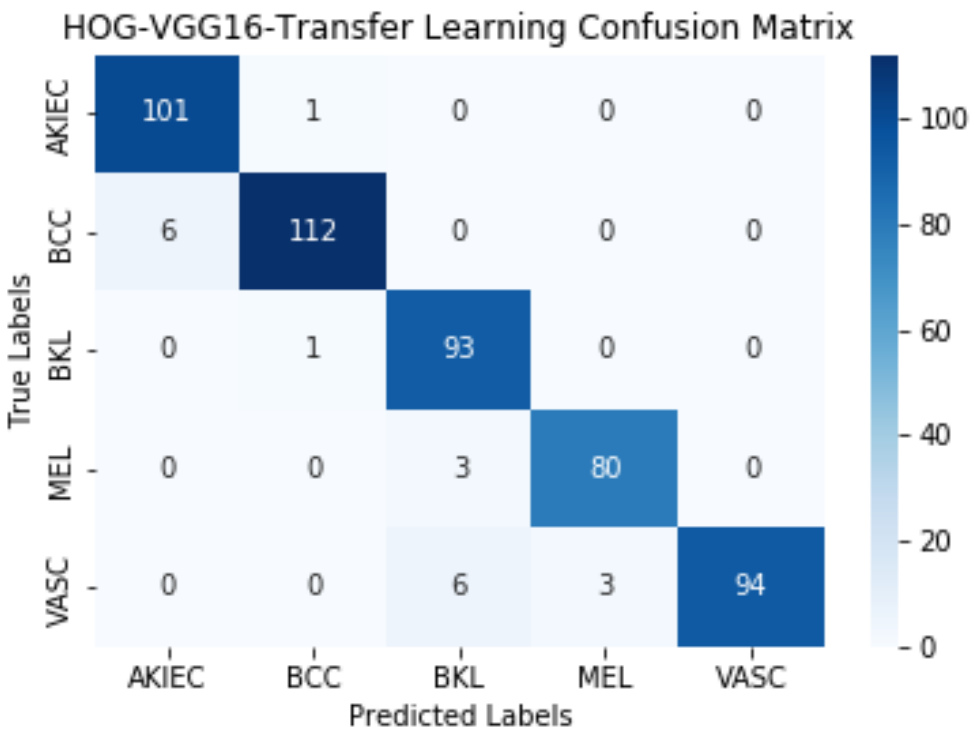


Figure 5. 11. Confusion matrix of HOG – VGG 16 with TL

5.12. The HOG – VGG 16 with TL training and validation curve result

Figure 5.12 illustrates the training and validation accuracy and loss curves over 40 epochs. It also the baseline VGG16 model and HOG + Transfer learning consistently demonstrated improved accuracy and reduced loss throughout the training process. This shows that the model can keep learning from the dataset all the way up to the 40th epoch. However, after this point, the model converged to a stable state.

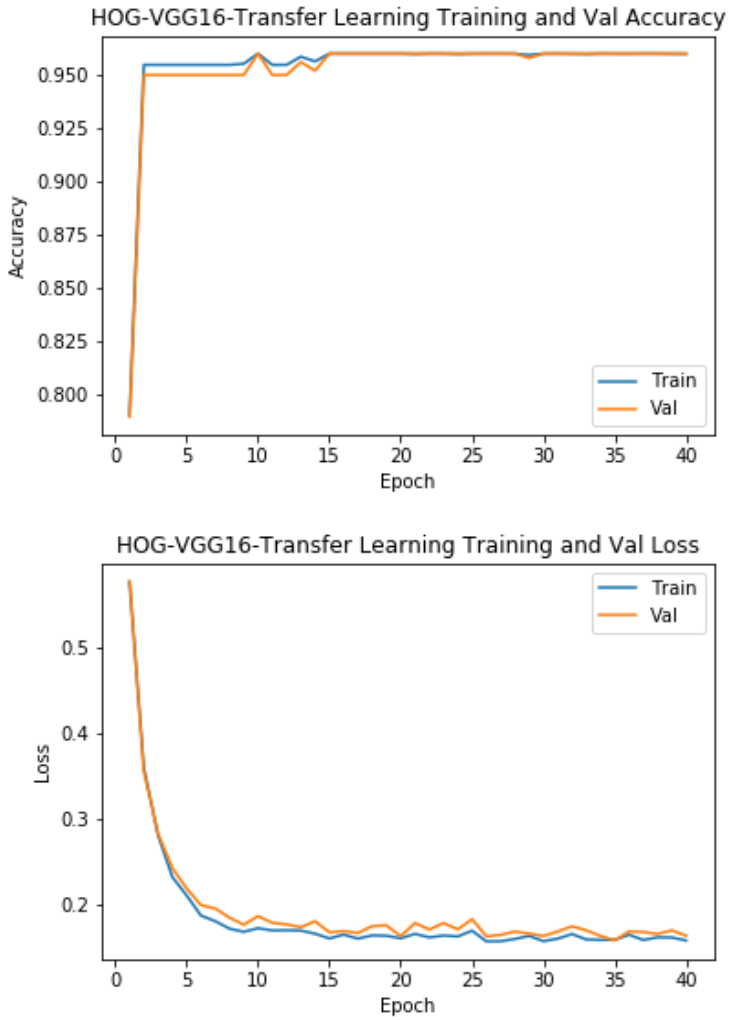


Figure 5. 12 the Val accuracy and Val Loss curve result

5.13. The skin cancer classification performance report

Performance metrics reports are essential tools for evaluating the effectiveness of various systems, models, or processes. They provide quantitative data that can be used to identify strengths, weaknesses, and areas for improvement.

When evaluating the classification performance of a HOG-VGG16 model, it's often useful to compare it to baseline methods, especially those based on transfer learning. These comparisons can provide insights into the effectiveness of the proposed approach. The table 5.1 is describes the all results of our proposed model.

Table 5.1 demonstrates that the combined use of VGG16, HOG features selection, and transfer learning significantly improved classification performance, achieving a 96% accuracy rate.

Table 5. 1 ***Classification performance of the proposed VGG16 with HOG and with TLand baseline methods***

Models	Accuracy	Precision	Recall	F1-score
VGG16 without HOG and without TL	0.906	0.904	0.907	0.902
VGG16 with HOG and without TL	0.950	0.953	0.953	0.950
VGG16 with HOG and with TL	**0.960**	**0.960**	**0.961**	**0.960**

5.14. Discussion

This study started out by addressing three research issues. Thus, the purpose of this section is to address those queries.

RQ1. Does feature selection strategy can help the improvement of deep learning model performance?

In our research, we deviated from traditional feature selection convolutional layers and employed the HOG Python module to identify high-value pixels within skin cancer training and validation images. Batches of 16 HOG-transformed images were then fed into a VGG16 backbone for classification. Our evaluation revealed a significant enhancement in classification accuracy, increasing from 90% without HOG to 95% with HOG. This demonstrates the substantial benefits of feature selection in improving skin cancer classification.

RQ2. How to design a deep learning model with feature selection strategy for the classification of skin cancer?

Existing research has not extensively explored the combined application of feature selection, CNNs, and transfer learning. Our study aims to investigate the potential of this approach for skin cancer image classification, which has shown promising results.

RQ3. Can we improve skin cancer classification performance via transfer learning and feature selection strategy?

In the proposed method, we employed end to end Histogram of Oriented Gradient (HOG), VGG16, and transfer learning jointly to improve classification performance. With proposed method we improved accuracy from 90% to 96%. And also we reduced confusions from 47/500 to 20/500.

CHAPTER SIX

CONCLUSIONS AND FUTURE WORKS

6.1. Conclusions

The CNN architecture's reliance on spatial context can hinder the learning of unique features. While the benefits of pre-trained weights have been explored, their combination with feature selection for skin cancer classification remains understudied. To address this, we propose a VGG16 model incorporating feature selection and transfer learning. Using the HAM10000 dermatoscopic image dataset from Kaggle, we mitigated class imbalance by removing highly skewed classes and employed data augmentation to expand the training set. The proposed model was evaluated using a variety of metrics, demonstrating superior classification performance compared to all baselines.

6.2. Future works

Our long-term objective is to improve our skin cancer classification system by adding more tagged skin lesions to our databases. This extension will make it easier to train a Deep Neural Network (DNN) that is more resilient and effective. To maximize classification accuracy and improve the model's predictive power, we will concentrate on optimizing pre-processing processes. Furthermore, in order to optimize the performance of our CNN, we also want to use optimization methods for hyper parameter tuning. The investigation of different optimization strategies has the potential to improve the model's performance even more.

To create a more generalized classification performance, future researchers can evaluate deeper CNNs, extract attention feature, and include patient history to investigate individual skin cancer cases. Furthermore, involving domain experts to collect, preprocess, train, validate and test local dataset.

References

Aggarwal, C. C. (2014). Educational and software resources for data classification. *Data Classification: Algorithms and Applications*, 657–665. https://doi.org/10.1201/b17320

Alam, T. M., Shaukat, K., Khan, W. A., Hameed, I. A., Almuqren, L. A., Raza, M. A., Aslam, M., & Luo, S. (2022). An Efficient Deep Learning-Based Skin Cancer Classifier for an Imbalanced Dataset. *Diagnostics*, *12*(9). https://doi.org/10.3390/diagnostics12092115

Ali, K., Shaikh, Z. A., Khan, A. A., & Laghari, A. A. (2022). Multiclass skin cancer classification using EfficientNets – a first step towards preventing skin cancer. *Neuroscience Informatics*, *2*(4), 100034. https://doi.org/10.1016/j.neuri.2021.100034

Ali, M. S., Miah, M. S., Haque, J., Rahman, M. M., & Islam, M. K. (2021a). An enhanced technique of skin cancer classification using deep convolutional neural network with transfer learning models. In *Machine Learning with Applications* (Vol. 5, p. 100036). https://doi.org/10.1016/j.mlwa.2021.100036

Ali, M. S., Miah, M. S., Haque, J., Rahman, M. M., & Islam, M. K. (2021b). An enhanced technique of skin cancer classification using deep convolutional neural network with transfer learning models. *Machine Learning with Applications*, *5*(February), 100036. https://doi.org/10.1016/j.mlwa.2021.100036

Anwar, S. M., Majid, M., Qayyum, A., Awais, M., Alnowami, M., & Khan, M. K. (2018). Medical Image Analysis using Convolutional Neural Networks: A Review. *Journal of Medical Systems*, *42*(11). https://doi.org/10.1007/s10916-018-1088-1

Aswir, & Misbah, H. (2018). No 主観的健康感を中心とした在宅高齢者における健康関連指標に関する共分散構造分析Title. *Photosynthetica*, *2*(1), 1–13. http://link.springer.com/10.1007/978-3-319-76887-8%0Ahttp://link.springer.com/10.1007/978-3-319-93594-2%0Ahttp://dx.doi.org/10.1016/B978-0-12-409517-5.00007-3%0Ahttp://dx.doi.org/10.1016/j.jff.2015.06.018%0Ahttp://dx.doi.org/10.1038/s41559-019-0877-3%0Aht

Beger, H. G., Rau, B., Gansauge, F., Leder, G., Schwarz, M., & Poch, B. (2008).

Bauchspeicheldrüsenkrebs - Heilungschancen minimal. In *Deutsches Arzteblatt* (Vol. 105, Issue 14, pp. 255–262). https://doi.org/10.3238/arztebl.2008.0255

Botunac, I., Panjkota, A., & Matetic, M. (2020). *g P er of Sy m po siu m kin g P ap Sy m po m*. https://doi.org/10.2507/31st.daaam.proceedings.xxx

Centers for Disease Control and Prevention. (2014). Melanoma Incidence and Mortality, United States—2012–2016. *USCS Data Brief*, *9*, 2014–2016.

Chen, X., Song, J., & Hilliges, O. (2019). Unpaired pose guided human image generation. *ArXiv*.

Coates, D. R., Chin, J. M., & Chung, S. T. L. (2011). 基因的改变NIH Public Access. *Bone*, *23*(1), 1–7. https://doi.org/10.1111/j.1600-0846.2007.00251.x.Unsupervised

De Benedetto, A., Kubo, A., & Beck, L. A. (2012). Skin barrier disruption: A requirement for allergen sensitization. *Journal of Investigative Dermatology*, *132*(3 PART 2), 949–963. https://doi.org/10.1038/jid.2011.435

Deng, L. (2012). The MNIST database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, *29*(6), 141–142. https://doi.org/10.1109/MSP.2012.2211477

Dermatology.pdf. (n.d.).

Erol, R., Bayraktar, M., Kockara, S., Kaya, S., & Halic, T. (2017). Texture based skin lesion abruptness quantification to detect malignancy. *BMC Bioinformatics*, *18*(Suppl 14). https://doi.org/10.1186/s12859-017-1892-5

Ghosh, A., Sufian, A., Sultana, F., Chakrabarti, A., & De, D. (2019a). Fundamental concepts of convolutional neural network. In *Intelligent Systems Reference Library* (Vol. 172, Issue January). https://doi.org/10.1007/978-3-030-32644-9_36

Ghosh, A., Sufian, A., Sultana, F., Chakrabarti, A., & De, D. (2019b). Fundamental concepts of convolutional neural network. In *Intelligent Systems Reference Library* (Vol. 172, Issue June). https://doi.org/10.1007/978-3-030-32644-9_36

Gonzalez, T. F. (2007). Handbook of approximation algorithms and metaheuristics. *Handbook of Approximation Algorithms and Metaheuristics*, 1–1432. https://doi.org/10.1201/9781420010749

Gutema, G., & Engidawork, E. (2018). Affordability of commonly prescribed antibiotics in a large tertiary teaching hospital in Ethiopia: A challenge for the national drug policy objective. *BMC Research Notes*, *11*(1), 4–9. https://doi.org/10.1186/s13104-018-4021-2

Hasan, M., Barman, S. Das, Islam, S., & Reza, A. W. (2019). Skin cancer detection using convolutional neural network. *ACM International Conference Proceeding Series*, 254–258. https://doi.org/10.1145/3330482.3330525

Hekler, A., Utikal, J. S., Enk, A. H., Hauschild, A., Weichenthal, M., Maron, R. C., Berking, C., Haferkamp, S., Klode, J., Schadendorf, D., Schilling, B., Holland-Letz, T., Izar, B., von Kalle, C., Fröhling, S., Brinker, T. J., Schmitt, L., Peitsch, W. K., Hoffmann, F., … Thiem, A. (2019). Superior skin cancer classification by the combination of human and artificial intelligence. *European Journal of Cancer*, *120*, 114–121. https://doi.org/10.1016/j.ejca.2019.07.019

HOG Convolutional neural networks and histogram-oriented gradients.pdf. (n.d.).

Humayun, M., Khalil, M. I., Alwakid, G., & Jhanjhi, N. Z. (2022). Superlative Feature Selection Based Image Classification Using Deep Learning in Medical Imaging. *Journal of Healthcare Engineering*, *2022*, 1–11. https://doi.org/10.1155/2022/7028717

Jasil, S. P. G., & Ulagamuthalvi, V. (2023). A hybrid CNN architecture for skin lesion classification using deep learning. *Soft Computing*, *0123456789*. https://doi.org/10.1007/s00500-023-08035-w

Johansson, M., Brodersen, J., Gøtzsche, P. C., & Jørgensen, K. J. (2019). Screening for reducing morbidity and mortality in malignant melanoma. *Cochrane Database of Systematic Reviews*, *2019*(6). https://doi.org/10.1002/14651858.CD012352.pub2

Karpathy, A. (2015). Deep Visual-Semantic Alignments for Generating Image Descriptions image regions , bidirectional Recurrent Neural Networks ture that uses the inferred alignments to learn to generate. *Cvpr*, 3128–3137.

Kuppam, S. R. (2020). The Skinny on Moisturizers: A Brief Report. *Dermatology – Open Journal*, *5*(1), 1–5. https://doi.org/10.17140/drmtoj-5-138

Miao, J., & Niu, L. (2016). A Survey on Feature Selection. *Procedia Computer Science*,

91(Itqm), 919–926. https://doi.org/10.1016/j.procs.2016.07.111

Patil, A., & Rane, M. (2021). Convolutional Neural Networks: An Overview and Its Applications in Pattern Recognition. *Smart Innovation, Systems and Technologies*, *195*, 21–30. https://doi.org/10.1007/978-981-15-7078-0_3

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., & Fei-Fei, L. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, *115*(3), 211–252. https://doi.org/10.1007/s11263-015-0816-y

Sam, G. (2012). Chapter 3 – Research Methodology and Research Method. *Research Methodology and Research Method*, *March 2012*, 43.

Sandeep Kumar, T., Annappa, B., & Dodia, S. (2023). Classification of Skin Cancer Images using Lightweight Convolutional Neural Network. *2023 4th International Conference for Emerging Technology, INCET 2023*. https://doi.org/10.1109/INCET57972.2023.10170637

Silpa, S. R., & V, C. (2013). a Review on Skin Cancer. *International Research Journal of Pharmacy*, *4*(8), 83–88. https://doi.org/10.7897/2230-8407.04814

Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 1–14.

Sultana, F., Sufian, A., & Dutta, P. (2018). Advancements in image classification using convolutional neural network. *Proceedings - 2018 4th IEEE International Conference on Research in Computational Intelligence and Communication Networks, ICRCICN 2018*, 122–129. https://doi.org/10.1109/ICRCICN.2018.8718718

Sunyoto, A., Pristyanto, Y., Setyanto, A., Alarfaj, F., Almusallam, N., & Alreshoodi, M. (2022). The Performance Evaluation of Transfer Learning VGG16 Algorithm on Various Chest X-ray Imaging Datasets for COVID-19 Classification. *International Journal of Advanced Computer Science and Applications*, *13*(9), 196–203. https://doi.org/10.14569/IJACSA.2022.0130923

Teuwen, J., & Moriakov, N. (2019). Convolutional neural networks. In *Handbook of Medical*

Image Computing and Computer Assisted Intervention. Elsevier Inc. https://doi.org/10.1016/B978-0-12-816176-0.00025-9

Tschandl, P., Rosendahl, C., & Kittler, H. (2018). The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *Scientific Data*, *5*(1), 180161. https://doi.org/10.1038/sdata.2018.161

Wadajo, C. W., & Science, A. (2023). *Cloud based Employee Timesheet Model Using Facial Recognition ; in case of Oromia Occupational Competence Assurance Agency*. *September*.

Walter, F. M., Prevost, A. T., Vasconcelos, J., Hall, P. N., Burrows, N. P., Morris, H. C., Kinmonth, A. L., & Emery, J. D. (2013). Using the 7-point checklist as a diagnostic aid for pigmented skin lesions in general practice: A diagnostic validation study. *British Journal of General Practice*, *63*(610). https://doi.org/10.3399/bjgp13X667213

Wang, T., Wu, D. J., & Ng, A. Y. (n.d.). *ICPR12-TextRecognitionConvNeuralNets.pdf*.

World Health Organization. (2014). WORLD CANCER REPORT.pdf. In *World Health Organization* (pp. 182–185). http://www.env.go.jp/air/asbestos/commi_hhmd/03/ext01.pdf

World Health Organization. (2018). Seychelles 2018 Update. *Monitoring Progress on Universal Health Coverage and the Health-Related Sustainable Development Goals in the South- East Asia Region 2018*.

Xu, Z., Sheykhahmad, F. R., Ghadimi, N., & Razmjooy, N. (2020). Computer-aided diagnosis of skin cancer based on soft computing techniques. *Open Medicine (Poland)*, *15*(1), 860–871. https://doi.org/10.1515/med-2020-0131

Yildirim, M., & Çinar, A. (2021). Classification of Skin Cancer Images with Convolutional Neural Network Architectures. *Turkish Journal of Science & Technology Research Paper*, *16*(2), 187–195.

APPENDEX

Appendix: A. The libraries used

```
from __future__ import print_function
from __future__ import division

import torch.nn as nn
from torchvision import datasets, models
import torchvision

import argparse
import csv
import os
import os.path as osp
import shutil
import time
from functools import partial
import tqdm
import torchvision.models as models
import numpy as np
import torch
import torch.nn as nn
import torch.optim as optim
from torch.autograd import Variable
from torch.backends import cudnn
from torch.optim.lr_scheduler import ReduceLROnPlateau
import pdb
import  DataPreProccessing, DataLoader
```

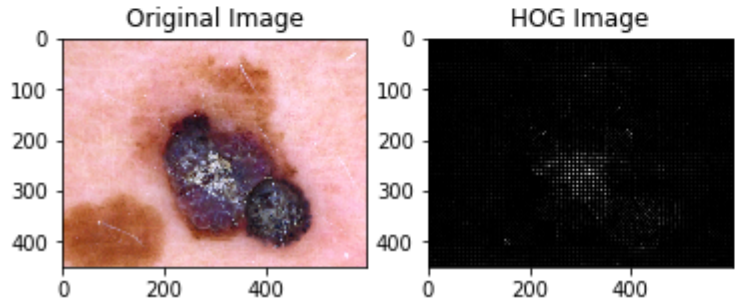
Appendix: B. HOG features calculator and displayed image

```
# Calculate the HOG features
hog_features, hog_image = hog(image_gray, visualize=True)

# Display the original image
plt.subplot(1, 2, 1)
plt.imshow(image, cmap='gray')
plt.title('Original Image')
plt.axis('on')

# Display the HOG image
plt.subplot(1, 2, 2)
plt.imshow(hog_image, cmap='gray')
plt.title('HOG Image')
plt.axis('on')

plt.show()
```



Appendix: C. HOG features calculator and preprocessing the images sample code

```
def HOG(img):
    # Apply CLAHE
    clahe = exposure.equalize_adapthist(image_gray, clip_limit=3.0)
    # Calculate the HOG features
    hog_features, hog_image = hog(clahe, visualize=True)
    # Apply gamma correction
    gamma_corrected_hog_image = exposure.adjust_gamma(hog_image, gamma=1.5)

def preproccessing():
    image_dir ="/HAM1000" #Generated image directory as trainingset
    imgFiles_train = [
        join(image_dir, fn)  # Create full paths to images
        for fn in os.listdir(image_dir)  # For each item in the image folder
        if isfile(join(image_dir, fn))  # If the item is indeed a file
          and fn.lower().endswith(('.png', '.jpg'))
        ]
```

```
# for train and validation  splitting
data_size_test = len(imgFiles_train)
indices_test = list(range(data_size_test))
split_test = int(np.floor(0.1 * data_size_test))
random_seed = 42
shuffle_datset = True
if shuffle_datset:
    torch.manual_seed(random_seed)
    np.random.shuffle(indices_test)
train_loader, valid_loader = indices_test[split_test:], indices_test[:split_test]
train_loader = SubsetRandomSampler(train_loader)
valid_loader = SubsetRandomSampler(valid_loader)
```

Appendix: D. Normalizing the images

```
## transforms.Normalize([0.485, 0.456, 0.406], [0.229, 0.224, 0.225])##
    train_loader = hog(torch.utils.data.DataLoader(
        DataLoader.ImageList(
            imgFiles_train, transform=transforms.Compose([transforms.ToTensor(),
transforms.Normalize((0.5, 0.5,
                                                          0.5), (0.5, 0.5,
                                                              0.5))])),
        batch_size=batch_size, num_workers=0, sampler=train_loader, drop_last=True,
pin_memory=True))
    valid_loader = hog(torch.utils.data.DataLoader(
        DataLoader.ImageList(imgFiles_train,
transform=transforms.Compose([transforms.ToTensor(), transforms.Normalize((0.5, 0.5,
                                                          0.5), (0.5, 0.5,
                                                              0.5))])),
                                                              batch_size=batch_size,
num_workers=0,
                                                              sampler=valid_loader,
drop_last=True,
                                                              pin_memory=True))
    imgFiles_test = [
        join(image_dir_test, fn)  # Create full paths to images
        for fn in os.listdir(image_dir_test)  # For each item in the image folder
```

```
    if isfile(join(image_dir_test, fn))  # If the item is indeed a file
      and fn.lower().endswith(('.png', '.jpg'))
    # Which ends with an image suffix (can add more types here if needed)
  ]
  test_list = imgFiles_test
  TestLoader = torch.utils.data.DataLoader(
    DataLoader.ImageList(test_list,
                         transform=transforms.Compose([transforms.ToTensor(),
transforms.Normalize((0.485, 0.485,
                                                    0.485), (0.485, 0.485,
                                                    0.485))])),
    batch_size=1, shuffle=True, num_workers=0, pin_memory=True,drop_last=True)
  return  train_loader, valid_loader
```

Appendix: E. The VGG 16 epoch sample training

```
Epoch: 1, Train Loss: 2.5125, Val Loss: 2.4590, Train Accuracy: 0.2086, Val Accuracy: 0.2060
Epoch: 2, Train Loss: 1.8569, Val Loss: 1.8289, Train Accuracy: 0.2604, Val Accuracy: 0.2380
Epoch: 3, Train Loss: 1.5954, Val Loss: 1.5929, Train Accuracy: 0.3261, Val Accuracy: 0.3300
Epoch: 4, Train Loss: 1.5245, Val Loss: 1.5351, Train Accuracy: 0.4660, Val Accuracy: 0.4500
Epoch: 5, Train Loss: 1.4853, Val Loss: 1.4977, Train Accuracy: 0.5428, Val Accuracy: 0.5100
Epoch: 6, Train Loss: 1.4447, Val Loss: 1.4564, Train Accuracy: 0.5982, Val Accuracy: 0.5600
Epoch: 7, Train Loss: 1.4033, Val Loss: 1.4154, Train Accuracy: 0.6368, Val Accuracy: 0.6000
Epoch: 8, Train Loss: 1.3610, Val Loss: 1.3728, Train Accuracy: 0.6720, Val Accuracy: 0.6460
Epoch: 9, Train Loss: 1.3187, Val Loss: 1.3296, Train Accuracy: 0.7021, Val Accuracy: 0.6760
Epoch: 10, Train Loss: 1.2764, Val Loss: 1.2861, Train Accuracy: 0.7476, Val Accuracy: 0.7280
Epoch: 11, Train Loss: 1.2342, Val Loss: 1.2445, Train Accuracy: 0.7694, Val Accuracy: 0.7380
Epoch: 12, Train Loss: 1.1922, Val Loss: 1.2014, Train Accuracy: 0.7807, Val Accuracy: 0.7500
Epoch: 13, Train Loss: 1.1505, Val Loss: 1.1593, Train Accuracy: 0.7880, Val Accuracy: 0.7660
Epoch: 14, Train Loss: 1.1092, Val Loss: 1.1185, Train Accuracy: 0.7900, Val Accuracy: 0.7640
Epoch: 15, Train Loss: 1.0686, Val Loss: 1.0773, Train Accuracy: 0.7984, Val Accuracy: 0.7720
Epoch: 16, Train Loss: 1.0284, Val Loss: 1.0371, Train Accuracy: 0.7960, Val Accuracy: 0.7680
Epoch: 17, Train Loss: 0.9890, Val Loss: 0.9974, Train Accuracy: 0.7996, Val Accuracy: 0.7720
Epoch: 18, Train Loss: 0.9504, Val Loss: 0.9590, Train Accuracy: 0.8010, Val Accuracy: 0.7720
Epoch: 19, Train Loss: 0.9126, Val Loss: 0.9201, Train Accuracy: 0.8051, Val Accuracy: 0.7760
Epoch: 20, Train Loss: 0.8761, Val Loss: 0.8848, Train Accuracy: 0.8070, Val Accuracy: 0.7740
Epoch: 21, Train Loss: 0.8407, Val Loss: 0.8470, Train Accuracy: 0.8103, Val Accuracy: 0.7800
Epoch: 22, Train Loss: 0.8069, Val Loss: 0.8125, Train Accuracy: 0.8138, Val Accuracy: 0.7820
Epoch: 23, Train Loss: 0.7744, Val Loss: 0.7801, Train Accuracy: 0.8148, Val Accuracy: 0.7800
Epoch: 24, Train Loss: 0.7437, Val Loss: 0.7482, Train Accuracy: 0.8263, Val Accuracy: 0.8000
Epoch: 25, Train Loss: 0.7145, Val Loss: 0.7189, Train Accuracy: 0.8254, Val Accuracy: 0.7980
Epoch: 26, Train Loss: 0.6871, Val Loss: 0.6910, Train Accuracy: 0.8310, Val Accuracy: 0.8020
Epoch: 27, Train Loss: 0.6615, Val Loss: 0.6660, Train Accuracy: 0.8394, Val Accuracy: 0.8060
Epoch: 28, Train Loss: 0.6377, Val Loss: 0.6400, Train Accuracy: 0.8396, Val Accuracy: 0.8140
Epoch: 29, Train Loss: 0.6156, Val Loss: 0.6180, Train Accuracy: 0.8458, Val Accuracy: 0.8140
Epoch: 30, Train Loss: 0.5953, Val Loss: 0.5986, Train Accuracy: 0.8560, Val Accuracy: 0.8320
Epoch: 31, Train Loss: 0.5766, Val Loss: 0.5794, Train Accuracy: 0.8588, Val Accuracy: 0.8320
Epoch: 32, Train Loss: 0.5594, Val Loss: 0.5610, Train Accuracy: 0.8681, Val Accuracy: 0.8480
Epoch: 33, Train Loss: 0.5436, Val Loss: 0.5452, Train Accuracy: 0.8729, Val Accuracy: 0.8480
Epoch: 34, Train Loss: 0.5294, Val Loss: 0.5303, Train Accuracy: 0.8760, Val Accuracy: 0.8520
Epoch: 35, Train Loss: 0.5163, Val Loss: 0.5177, Train Accuracy: 0.8803, Val Accuracy: 0.8600
Epoch: 36, Train Loss: 0.5043, Val Loss: 0.5045, Train Accuracy: 0.8924, Val Accuracy: 0.8740
Epoch: 37, Train Loss: 0.4936, Val Loss: 0.4936, Train Accuracy: 0.8964, Val Accuracy: 0.8820
Epoch: 38, Train Loss: 0.4838, Val Loss: 0.4841, Train Accuracy: 0.8974, Val Accuracy: 0.8760
Epoch: 39, Train Loss: 0.4748, Val Loss: 0.4747, Train Accuracy: 0.9051, Val Accuracy: 0.8920
Epoch: 40, Train Loss: 0.4666, Val Loss: 0.4661, Train Accuracy: 0.9189, Val Accuracy: 0.9060
```

Appendix: F. The VGG 16 + HOG epoch sample training

```
Epoch: 1, Train Loss: 1.8548, Val Loss: 1.8216, Train Accuracy: 0.2545, Val Accuracy: 0.2700
Epoch: 2, Train Loss: 1.5364, Val Loss: 1.5230, Train Accuracy: 0.4984, Val Accuracy: 0.5480
Epoch: 3, Train Loss: 1.4766, Val Loss: 1.4647, Train Accuracy: 0.5681, Val Accuracy: 0.6400
Epoch: 4, Train Loss: 1.4160, Val Loss: 1.4019, Train Accuracy: 0.6124, Val Accuracy: 0.6640
Epoch: 5, Train Loss: 1.3512, Val Loss: 1.3314, Train Accuracy: 0.6780, Val Accuracy: 0.7140
Epoch: 6, Train Loss: 1.2816, Val Loss: 1.2619, Train Accuracy: 0.7169, Val Accuracy: 0.7460
Epoch: 7, Train Loss: 1.2071, Val Loss: 1.1859, Train Accuracy: 0.7465, Val Accuracy: 0.7720
Epoch: 8, Train Loss: 1.1290, Val Loss: 1.1029, Train Accuracy: 0.7702, Val Accuracy: 0.7960
Epoch: 9, Train Loss: 1.0480, Val Loss: 1.0239, Train Accuracy: 0.7702, Val Accuracy: 0.7960
Epoch: 10, Train Loss: 0.9672, Val Loss: 0.9419, Train Accuracy: 0.7702, Val Accuracy: 0.7960
Epoch: 11, Train Loss: 0.8888, Val Loss: 0.8621, Train Accuracy: 0.7702, Val Accuracy: 0.7960
Epoch: 12, Train Loss: 0.8144, Val Loss: 0.7909, Train Accuracy: 0.7702, Val Accuracy: 0.7960
Epoch: 13, Train Loss: 0.7465, Val Loss: 0.7266, Train Accuracy: 0.7702, Val Accuracy: 0.7960
Epoch: 14, Train Loss: 0.6855, Val Loss: 0.6664, Train Accuracy: 0.7702, Val Accuracy: 0.7960
Epoch: 15, Train Loss: 0.6328, Val Loss: 0.6166, Train Accuracy: 0.7746, Val Accuracy: 0.7980
Epoch: 16, Train Loss: 0.5888, Val Loss: 0.5757, Train Accuracy: 0.8021, Val Accuracy: 0.8140
Epoch: 17, Train Loss: 0.5516, Val Loss: 0.5420, Train Accuracy: 0.8631, Val Accuracy: 0.8760
Epoch: 18, Train Loss: 0.5216, Val Loss: 0.5108, Train Accuracy: 0.7702, Val Accuracy: 0.7960
Epoch: 19, Train Loss: 0.4964, Val Loss: 0.4890, Train Accuracy: 0.8578, Val Accuracy: 0.8620
Epoch: 20, Train Loss: 0.4762, Val Loss: 0.4718, Train Accuracy: 0.9347, Val Accuracy: 0.9360
Epoch: 21, Train Loss: 0.4591, Val Loss: 0.4536, Train Accuracy: 0.8967, Val Accuracy: 0.8980
Epoch: 22, Train Loss: 0.4449, Val Loss: 0.4432, Train Accuracy: 0.9547, Val Accuracy: 0.9500
Epoch: 23, Train Loss: 0.4322, Val Loss: 0.4289, Train Accuracy: 0.9160, Val Accuracy: 0.9100
Epoch: 24, Train Loss: 0.4214, Val Loss: 0.4174, Train Accuracy: 0.9086, Val Accuracy: 0.9060
Epoch: 25, Train Loss: 0.4111, Val Loss: 0.4104, Train Accuracy: 0.9547, Val Accuracy: 0.9500
Epoch: 26, Train Loss: 0.4028, Val Loss: 0.4039, Train Accuracy: 0.9547, Val Accuracy: 0.9500
Epoch: 27, Train Loss: 0.3935, Val Loss: 0.3937, Train Accuracy: 0.9547, Val Accuracy: 0.9500
Epoch: 28, Train Loss: 0.3856, Val Loss: 0.3867, Train Accuracy: 0.9547, Val Accuracy: 0.9500
Epoch: 29, Train Loss: 0.3779, Val Loss: 0.3770, Train Accuracy: 0.9547, Val Accuracy: 0.9500
Epoch: 30, Train Loss: 0.3700, Val Loss: 0.3693, Train Accuracy: 0.9547, Val Accuracy: 0.9500
Epoch: 31, Train Loss: 0.3622, Val Loss: 0.3622, Train Accuracy: 0.9547, Val Accuracy: 0.9500
Epoch: 32, Train Loss: 0.3550, Val Loss: 0.3565, Train Accuracy: 0.9547, Val Accuracy: 0.9500
Epoch: 33, Train Loss: 0.3471, Val Loss: 0.3488, Train Accuracy: 0.9547, Val Accuracy: 0.9500
Epoch: 34, Train Loss: 0.3395, Val Loss: 0.3407, Train Accuracy: 0.9547, Val Accuracy: 0.9500
Epoch: 35, Train Loss: 0.3318, Val Loss: 0.3335, Train Accuracy: 0.9547, Val Accuracy: 0.9500
Epoch: 36, Train Loss: 0.3247, Val Loss: 0.3271, Train Accuracy: 0.9547, Val Accuracy: 0.9500
Epoch: 37, Train Loss: 0.3169, Val Loss: 0.3188, Train Accuracy: 0.9547, Val Accuracy: 0.9500
Epoch: 38, Train Loss: 0.3094, Val Loss: 0.3121, Train Accuracy: 0.9547, Val Accuracy: 0.9500
Epoch: 39, Train Loss: 0.3017, Val Loss: 0.3041, Train Accuracy: 0.9547, Val Accuracy: 0.9500
Epoch: 40, Train Loss: 0.2945, Val Loss: 0.2974, Train Accuracy: 0.9547, Val Accuracy: 0.9500
```

Appendix: G. The VGG 16 + HOG + TL epoch training sample

```
Epoch: 1, Train Loss: 0.5769, Val Loss: 0.5784, Train Accuracy: 0.7911, Val Accuracy: 0.7900
Epoch: 2, Train Loss: 0.3598, Val Loss: 0.3555, Train Accuracy: 0.9547, Val Accuracy: 0.9500
Epoch: 3, Train Loss: 0.2804, Val Loss: 0.2824, Train Accuracy: 0.9547, Val Accuracy: 0.9500
Epoch: 4, Train Loss: 0.2317, Val Loss: 0.2423, Train Accuracy: 0.9547, Val Accuracy: 0.9500
Epoch: 5, Train Loss: 0.2102, Val Loss: 0.2185, Train Accuracy: 0.9547, Val Accuracy: 0.9500
Epoch: 6, Train Loss: 0.1868, Val Loss: 0.1989, Train Accuracy: 0.9547, Val Accuracy: 0.9500
Epoch: 7, Train Loss: 0.1802, Val Loss: 0.1950, Train Accuracy: 0.9547, Val Accuracy: 0.9500
Epoch: 8, Train Loss: 0.1714, Val Loss: 0.1843, Train Accuracy: 0.9547, Val Accuracy: 0.9500
Epoch: 9, Train Loss: 0.1678, Val Loss: 0.1761, Train Accuracy: 0.9553, Val Accuracy: 0.9500
Epoch: 10, Train Loss: 0.1719, Val Loss: 0.1861, Train Accuracy: 0.9600, Val Accuracy: 0.9600
Epoch: 11, Train Loss: 0.1693, Val Loss: 0.1782, Train Accuracy: 0.9547, Val Accuracy: 0.9500
Epoch: 12, Train Loss: 0.1695, Val Loss: 0.1764, Train Accuracy: 0.9547, Val Accuracy: 0.9500
Epoch: 13, Train Loss: 0.1692, Val Loss: 0.1731, Train Accuracy: 0.9586, Val Accuracy: 0.9560
Epoch: 14, Train Loss: 0.1655, Val Loss: 0.1800, Train Accuracy: 0.9564, Val Accuracy: 0.9520
Epoch: 15, Train Loss: 0.1599, Val Loss: 0.1671, Train Accuracy: 0.9600, Val Accuracy: 0.9600
Epoch: 16, Train Loss: 0.1644, Val Loss: 0.1685, Train Accuracy: 0.9600, Val Accuracy: 0.9600
Epoch: 17, Train Loss: 0.1598, Val Loss: 0.1663, Train Accuracy: 0.9600, Val Accuracy: 0.9600
Epoch: 18, Train Loss: 0.1635, Val Loss: 0.1744, Train Accuracy: 0.9600, Val Accuracy: 0.9600
Epoch: 19, Train Loss: 0.1631, Val Loss: 0.1752, Train Accuracy: 0.9600, Val Accuracy: 0.9600
Epoch: 20, Train Loss: 0.1602, Val Loss: 0.1627, Train Accuracy: 0.9600, Val Accuracy: 0.9600
Epoch: 21, Train Loss: 0.1653, Val Loss: 0.1780, Train Accuracy: 0.9598, Val Accuracy: 0.9600
Epoch: 22, Train Loss: 0.1612, Val Loss: 0.1703, Train Accuracy: 0.9600, Val Accuracy: 0.9600
Epoch: 23, Train Loss: 0.1633, Val Loss: 0.1779, Train Accuracy: 0.9600, Val Accuracy: 0.9600
Epoch: 24, Train Loss: 0.1623, Val Loss: 0.1706, Train Accuracy: 0.9598, Val Accuracy: 0.9600
Epoch: 25, Train Loss: 0.1691, Val Loss: 0.1825, Train Accuracy: 0.9600, Val Accuracy: 0.9600
Epoch: 26, Train Loss: 0.1566, Val Loss: 0.1624, Train Accuracy: 0.9600, Val Accuracy: 0.9600
Epoch: 27, Train Loss: 0.1567, Val Loss: 0.1641, Train Accuracy: 0.9600, Val Accuracy: 0.9600
Epoch: 28, Train Loss: 0.1593, Val Loss: 0.1679, Train Accuracy: 0.9600, Val Accuracy: 0.9600
Epoch: 29, Train Loss: 0.1629, Val Loss: 0.1656, Train Accuracy: 0.9594, Val Accuracy: 0.9580
Epoch: 30, Train Loss: 0.1568, Val Loss: 0.1629, Train Accuracy: 0.9600, Val Accuracy: 0.9600
Epoch: 31, Train Loss: 0.1597, Val Loss: 0.1681, Train Accuracy: 0.9600, Val Accuracy: 0.9600
Epoch: 32, Train Loss: 0.1651, Val Loss: 0.1740, Train Accuracy: 0.9599, Val Accuracy: 0.9600
Epoch: 33, Train Loss: 0.1588, Val Loss: 0.1695, Train Accuracy: 0.9598, Val Accuracy: 0.9600
Epoch: 34, Train Loss: 0.1582, Val Loss: 0.1628, Train Accuracy: 0.9600, Val Accuracy: 0.9600
Epoch: 35, Train Loss: 0.1583, Val Loss: 0.1577, Train Accuracy: 0.9600, Val Accuracy: 0.9600
Epoch: 36, Train Loss: 0.1643, Val Loss: 0.1682, Train Accuracy: 0.9599, Val Accuracy: 0.9600
Epoch: 37, Train Loss: 0.1581, Val Loss: 0.1674, Train Accuracy: 0.9600, Val Accuracy: 0.9600
Epoch: 38, Train Loss: 0.1613, Val Loss: 0.1652, Train Accuracy: 0.9600, Val Accuracy: 0.9600
Epoch: 39, Train Loss: 0.1610, Val Loss: 0.1694, Train Accuracy: 0.9599, Val Accuracy: 0.9600
Epoch: 40, Train Loss: 0.1576, Val Loss: 0.1631, Train Accuracy: 0.9599, Val Accuracy: 0.9600
```