SAVITRIBAI PHULE PUNE UNIVERSITY

A PRELIMINARY PROJECT REPORT ON

Early detection of Skin Cancer using deep learning techniques.

SUBMITTED TOWARDS THE
PARTIAL FULFILLMENT OF THE REQUIREMENTS OF

BACHELOR OF ENGINEERING (Computer Engineering)

BY

Vedant Talele	Exam No:B191224338
Trisha Thakur	Exam No:B191224340
Aditya Ubale	Exam No:B191224341
Yash Kadam	Exam No:B191224258

Under The Guidance of

Prof. Shailaja Lohar

DEPARTMENT OF COMPUTER ENGINEERING
Pimpri Chinchwad College of Engineering And Research
Ravet

A.Y.2023-2024

Pimpri Chinchwad College Of Engineering And Research
DEPARTMENT OF COMPUTER ENGINEERING

CERTIFICATE

This is to certify that the Project Entitled

Early detection of Skin Cancer using deep learning techniques.

Submitted by

Vedant Talele	Exam No:B191224338
Trisha Thakur	Exam No:B191224340
Aditya Ubale	Exam No:B191224341
Yash Kadam	Exam No:B191224258

is a bonafide work carried out by Students under the supervision of Prof. Shailaja Lohar and it is submitted towards the partial fulfillment of the requirement of Bachelor of Engineering (Computer Engineering) Project.

Prof. Shailaja Lohar Internal Guide Dept. of Computer Engg.	Dr. Archana Chaugule H.O.D Dept. of Computer Engg.
Dr. Abhijeet Jadhav Project Co-ordinator	Prof.Dr. H.U.Tiwari Principal

EXTERNAL SIGNATURE

Abstract

Skin cancer is considered as one of the most prevalent and potentially life-threatening forms of cancer and there is a drastic increase in the rate of deaths due to lack of knowledge on the symptoms and their prevention. Thus, early detection at premature stage is necessary so that one can prevent the spreading of cancer. The primary objective of this project is to develop a robust deep learning model capable of accurately predicting different types of skin cancer from images. With a dataset of skin images, containing various skin cancer types and their subclasses, the model will undergo training, validation, and testing phases. Convolutional Neural Networks (CNNs), known for their exceptional image recognition capabilities, will serve as the foundation of our model. Along with CNN, we employ various machine learning algorithms like KNN, SVM, and Random Forest, and we're developing a special combined method to improve our predictions. The scope of this project extends to the challenges posed by variations in skin images such as size, color, texture, and lighting conditions. By extracting intricate features from the images, the model aims to capture subtle patterns indicative of various skin cancer types. Moreover, an emphasis will be placed on interpretability, allowing dermatologists and medical practitioners to comprehend the model's decision-making process.

Acknowledgments

It gives us great pleasure in presenting the preliminary project report on ***'Early detection of Skin Cancer using deep learning techniques.'****.*

I would like to take this opportunity to thank my internal guide ***Prof. Shailaja Lohar*** *for giving me all the help and guidance I needed. I am really grateful to them for their kind support. Their valuable suggestions were very helpful.*

I am also grateful to ***Dr. Archana Chaugule****, Head of Computer Engineering Department, CollegeName for his indispensable support, suggestions.*

In the end our special thanks to ***Prof. Shailaja Lohar*** *for providing various resources such as laboratory with all needed software platforms, continuous Internet connection, for Our Project.*

Vedant Talele
Trisha Thakur
Aditya Ubale
Yash Kadam
(B.E. Computer Engg.)

INDEX

List of Figures

List of Tables

CHAPTER 1

SYNOPSIS

1.1 PROJECT TITLE

Early detection of Skin Cancer using deep learning techniques.

1.2 PROJECT OPTION

Internal Project

1.3 INTERNAL GUIDE

Prof. Shailaja Lohar

1.4 SPONSORSHIP AND EXTERNAL GUIDE

1.5 TECHNICAL KEYWORDS (AS PER ACM KEYWORDS)

Technical Key Words:

- Deep Learning
- Convolutional Neural Networks (CNNs)
- Machine Learning Algorithms
- Dermoscopic Images

1.6 PROBLEM STATEMENT

Skin cancer poses a significant threat, with rising mortality rates due to inadequate awareness and early detection. Our challenge is to develop a hybrid deep learning-based system capable of accurately identifying various skin cancer types from dermoscopic images and making it interpretable to dermatologists and treat patients in the premature stage of cancer.

1.7 ABSTRACT

Skin cancer is considered as one of the most prevalent and potentially life-threatening forms of cancer and there is a drastic increase in the rate of deaths due to lack of

knowledge on the symptoms and their prevention. Thus, early detection at premature stage is necessary so that one can prevent the spreading of cancer. The primary objective of this project is to develop a robust deep learning model capable of accurately predicting different types of skin cancer from images. With a dataset of skin images, containing various skin cancer types and their subclasses, the model will undergo training, validation, and testing phases. Convolutional Neural Networks (CNNs), known for their exceptional image recognition capabilities, will serve as the foundation of our model. Along with CNN, we employ various machine learning algorithms like KNN, SVM, and Random Forest, and we're developing a special combined method to improve our predictions. The scope of this project extends to the challenges posed by variations in skin images such as size, color, texture, and lighting conditions. By extracting intricate features from the images, the model aims to capture subtle patterns indicative of various skin cancer types. Moreover, an emphasis will be placed on interpretability, allowing dermatologists and medical practitioners to comprehend the model's decision-making process.

1.8 GOALS AND OBJECTIVES

- The project aims to develop a system that can accurately detect skin cancer at an early stage.
- The system will use deep learning techniques to analyze dermoscopic images.
- The system will be able to predict the specific type of skin cancer present in a given image irrespective of the noise in that image.

1.9 RELEVANT MATHEMATICS ASSOCIATED WITH THE PROJECT

The following terms shows in detail working of project.

$S=\{s, e, X, Y, \varphi\}$

Where,

s = Start of the program.

1. Upload the image.

e = End of the program.

User can interpret the results from the screen.

X = Input of the program.

Input should be File which is uploaded by user.

Y = Output of the program.

User can interpret results based on the processing of the model.

φ = Failures and Success conditions.

- Success Conditions:

  1. The model must achieve a high accuracy in classifying skin cancer images. The accuracy should be comparable to or better than the accuracy of human dermatologists.
  2. The model must be able to classify different types of skin cancer.
  3. The model must be able to work with a variety of dermoscopic images, including images with different lighting conditions, resolutions, and backgrounds.
  4. The model must be robust to noise and other artifacts that may be present in dermoscopic images.

- Failure Conditions:

  1. The model may not achieve a high accuracy in classifying skin cancer images due to a number of factors, such as a small or low-quality training dataset.
  2. The model may not be able to work with all types of dermoscopic images because some images may be of poor quality or may contain noise or other artifacts that make them difficult for the model to interpret.

1.10 NAMES OF CONFERENCES / JOURNALS WHERE PAPERS CAN BE PUBLISHED

- IJARCSMS(International Journal of Advance Research in Computer Science And Management Studies)
- IJTRA(International Journal of Technical Research and Applications)
- IEEE(Institute of Electrical and Electronics Engineers)
- PGCON(Post Graduate Conference of Computer Engineering)

1.11 REVIEW OF CONFERENCE/JOURNAL PAPERS SUPPORTING PROJECT IDEA

- Aljohani and Turki (2022): This paper focuses on automatic classification of melanoma skin cancer using various deep learning models, including DenseNet201, MobileNetV2, ResNet, VGG, and GoogleNet. It is relevant as it addresses the use of deep learning for skin cancer classification. However, it highlights the importance of achieving higher test accuracy, which aligns with the project's objective of improving accuracy in early skin cancer detection.
- Gajera et al. (2022): This paper utilizes pretrained CNN models for the classification of dermoscopy images, acknowledging that using deep architectures on small datasets may lead to overfitting. It is relevant because it explores the use of CNNs for skin cancer classification and highlights the challenges of working with limited data. This is in line with the project's focus on image classification with deep learning.
- Bechelli and Delhommelle (2022): The paper evaluates the performance of different machine learning and deep learning algorithms on skin cancer datasets. It mentions the limitations in achieving high accuracy with deep learning models. It is relevant as it discusses the performance of deep learning algorithms in skin cancer detection. The challenges and limitations highlighted align with the project's goals of improving accuracy in skin cancer detection.

- Adegun et al. (2021): The paper proposes a method for skin lesion segmentation using a fully convolutional neural network. It mentions dataset challenges and the complexity of handling multiple classes. This paper is relevant as it addresses the segmentation of skin lesions, which is a crucial step in skin cancer detection. It relates to the project's methodology of image segmentation for skin cancer analysis.

- Inthiyaz et al.: Used pre-trained model for feature extraction and classification was performed using softmax classifier. This work was tested on a very small dataset; these results can not be generalized on large datasets.Inthiyaz et al. achieved an AUC of 0.87, which can still be improved.

- Qasim Gilani et al.: Used spiking neural network (SNN) for the classification of skin cancer images. The proposed spiking-VGG13 performed better, butthe specificity and precision of VGG-13 were higher.

- Khan et al(2021): Proposed CNN-based fully automated method for the classification and segmentation of images. The proposed model gave the best segmentation performance on PH2 , which has only 200 images; the effectiveness of the proposed method should be evaluated by testing it on larger datasets.

- Shinde et al. (2022): Lightweight model was proposed for the classification of skin cancer images on IOT devices. The proposed model in this work had lower sensitivity and specificity than other baseline models.

- Alenezi et al.(2022): Multi-stage deep model was used for the extraction of features from skin cancer images. The proposed work has limitations in terms of the time required for the parameter selection of the SVM classifier.

1.12 PLAN OF PROJECT EXECUTION

Using planner or alike project management tool.

CHAPTER 2

TECHNICAL KEYWORDS

2.1 AREA OF PROJECT

The project aims to tackle the pressing issue of skin cancer by harnessing deep learning techniques. It recognizes the significance of early detection in combatting this potentially life-threatening condition. To achieve its objectives, the project embarks on a comprehensive journey, starting with a literature survey. It delves into existing research, studying methodologies, and shedding light on certain limitations in current skin cancer detection models. Various machine learning and deep learning algorithms, including CNNs, are explored in this endeavor. The core aim is to build a robust system that can accurately detect different types of skin cancer from images, especially in their early stages. The focus is on dermoscopic images, and the project outlines the need for proper data acquisition, image segmentation, and the use of Python-based libraries for analysis. The project involves a well-structured work breakdown, and it aspires to bring a positive impact by enhancing the accuracy of skin cancer detection while making the process more accessible and reducing the burden on dermatologists. It represents a significant step towards leveraging technology for healthcare improvement.

2.2 TECHNICAL KEYWORDS

Technical Key Words:

- Deep Learning :
  Deep learning is a subset of machine learning that has gained significant prominence in recent years for its remarkable ability to automatically learn and extract features from data. It has revolutionized various fields, including image recognition, natural language processing, and even healthcare. In the context of the project on early skin cancer detection, deep learning techniques are employed to create a robust and highly accurate model for analyzing dermoscopic images. Deep learning models are designed to mimic the structure and function of the human brain by using artificial neural networks with multiple layers. These neural networks are capable of discovering intricate patterns and representations within the data. In the case of skin cancer detection, they

can identify nuanced features and textures in dermoscopic images, which are crucial for accurate diagnosis.

- Convolutional Neural Networks (CNNs):

  Convolutional Neural Networks (CNNs) are a class of deep learning models that have demonstrated exceptional effectiveness in image recognition and processing tasks. In the context of early skin cancer detection, CNNs serve as the foundation of the proposed system for analyzing dermoscopic images. CNNs are designed to process and understand visual data, making them highly suitable for tasks where features and patterns in images need to be extracted and classified. They are composed of multiple layers, including convolutional layers, pooling layers, and fully connected layers. Convolutional layers are particularly adept at identifying local features in an image, such as edges, textures, and shapes. For the skin cancer detection project, CNNs are trained on a dataset of dermoscopic images that contain various types and subclasses of skin cancer. During training, these networks learn to recognize distinctive features and patterns associated with different cancer types, enabling them to make accurate predictions on new, unseen images.

- Machine Learning Algorithms:

  Machine learning algorithms are the backbone of the project's early skin cancer detection system. These algorithms are responsible for the classification of skin lesions in dermoscopic images, providing the ability to distinguish between various types and subclasses of skin cancer. The choice of machine learning algorithms is crucial for the success of the project. In this context, machine learning techniques, especially deep learning, are selected for their capability to automatically learn patterns, features, and representations from the training data. This is essential for the accurate identification of skin cancer, which may present diverse visual characteristics in different individuals. Machine learning algorithms used in the project include convolutional neural networks (CNNs), a type of deep learning model that excels in image recognition. CNNs are particularly adept at identifying features and patterns in images, which makes them ideal for analyzing dermoscopic images.

- Dermoscopic Images:

  Dermoscopic images are a central component of the project for early skin cancer detection using deep learning techniques. These images provide the visual data necessary to train and test the machine learning algorithms, enabling the system to identify and classify different types of skin cancer accurately. Dermoscopic images are high-quality photographs of skin lesions, often captured using specialized dermatoscopes or digital cameras. These images are characterized by their high resolution and detail, which is crucial for the accurate diagnosis of skin cancer. The images may encompass various skin cancer types, and each image serves as a representation of a specific case.

CHAPTER 3

INTRODUCTION

3.1 PROJECT IDEA

Skin cancer is a prevalent and potentially life-threatening disease that has been on the rise due to a lack of awareness regarding its symptoms and prevention. Timely detection is essential to prevent its spread and save lives. This project, presented for the Bachelor's Degree in Computer Engineering, aims to address this critical healthcare challenge. The primary goal of this project is to develop a robust deep learning model capable of accurately identifying various types of skin cancer from dermoscopic images. These images contain diverse skin cancer types and subclasses. The system will undergo a rigorous process of training, validation, and testing, and Convolutional Neural Networks (CNNs) will serve as the foundation of this model. A thorough literature survey has been conducted, highlighting the methodologies and limitations of previous works in the field. Some of the limitations observed include low accuracy, especially with small datasets, and challenges related to overfitting. The proposed system will incorporate image acquisition, preprocessing, and classification. Python will be the primary language for implementation, utilizing libraries like OpenCV, Scikit-learn, and PyTorch. The work breakdown structure outlines the project's activities, including data collection, model selection, testing, and evaluation. The project will conclude with a comprehensive report. This project seeks to leverage deep learning and CNNs to create a valuable tool for the early detection of skin cancer. By doing so, it aims to enhance patient care, reduce the burden on healthcare professionals, and make skin cancer detection more accessible, ultimately contributing to improved healthcare outcomes.

3.2 MOTIVATION OF THE PROJECT

The motivation behind this project lies in addressing the pressing issue of skin cancer, which is one of the most prevalent and life-threatening forms of cancer. The alarming increase in mortality rates due to a lack of awareness about symptoms and preventive measures highlights the urgency for early detection. Traditional methods have proven insufficient, making it imperative to harness the power of deep learning techniques, specifically Convolutional Neural Networks (CNNs). This project aims to develop a robust system capable of accurately identifying various types of skin cancer from dermoscopic images. The ultimate goal is to empower medical professionals with a tool that can significantly enhance early diagnosis and, subsequently, patient care and survival rates.

3.3 LITERATURE SURVEY

Table 3.1: Research Papers on Skin Cancer Detection

Research article (Author/Year)	Objective/Proposed work	Methods/Techniques	Datasets	Relevant findings/Limitations identified
Inthiyaz et al.	Used pre-trained model for feature extraction and classification was performed using softmax classifier.	CNN	Xiangya-Derm	This work was tested on a very small dataset; these results can not be generalized on large datasets.Inthiyaz et al. achieved an AUC of 0.87, which can still be improved
Aljohani and Turki (2022)	Automatic Classification of Melanoma Skin Cancer with Deep Convolutional Neural Networks	DenseNet201, MobileNetV2, ResNet50V2, ResNet152V2, Xception, VGG16, VGG19, and GoogleNet	ISIC 2019	The maximum test accuracy of 76.09% was achieved using GoogleNet, which was quite low, and the model was tested only for the binary classification case.
Gajera et al.(2022)	Used eight pre-trained CNN models for the classification of dermoscopy images.	CNN model	ISIC 2016, and ISIC 2017 with only 200, 900, and 2000 training images.	Using deep architectures such as DenseNet-121 on small datasets may result in overfitting. Classification performance on the HAM10000 dataset was low.
Alenezi et al. (2022)	Multi-stage deep model was used for the extraction of features from skin	Relief algorithm	ISIC-2019 and ISIC-2020)	the proposed work has limitations in terms of the time required for the

	cancer images.			parameter selection of the SVM classifier
Shinde et al. (2022)	Lightweight model was proposed for the classification of skin cancer images on IOT devices	***Squeeze-MNet Model***	ISIC	The proposed model in this work had lower sensitivity and specificity than other baseline models.
Bechelli and Delhommelle(2022)	Performance of different machine learning and deep learning algorithms was evaluated on skin cancer datasets.	CNN, pre-trained VGG-16, Xception, ResNet50	Kaggle dataset, HAM10000	The deep learning models did not perform well on the HAM10000 dataset, with the best-performing architecture ,accuracy was low than the accuracy achieved on the ISIC dataset which had fewer images.
Khan et al(2021)	Proposed CNN-based fully automated method for the classification and segmentation of images.	ResNet101, DenseNet201	Segmentation (ISBI 2016, ISBI 2017, ISIC 2018, PH2), classification (HAM10000)	The proposed model gave the best segmentation performance on PH2 , which has only 200 images; the effectiveness of the proposed method should be evaluated by testing it on larger datasets.
Adegun et al. (2021)	proposed a method for skin lesion segmentation based on a fully convolutional neural network.	fully convolutional neural network	ISBI 2017, PH2	HAM1000 dataset consists of seven classes; they did not mention how they selected a subset of HAM10000 or how they prepared the dataset, as the confusion matrix only has three classes.
Qasim Gilani et al.	Used spiking neural network (SNN) for the classification of skin cancer images.	Spiking VGG-13	HAM10000	The proposed spiking-VGG13 performed better, butthe specificity and precision of VGG-13 were higher.

CHAPTER 4

PROBLEM DEFINITION AND SCOPE

4.1 PROBLEM STATEMENT

The problem statement revolves around the challenge of enhancing the accuracy of skin cancer detection using deep learning techniques. Researchers have explored various methodologies, but there are limitations, such as low accuracy, overfitting, and dataset selection concerns. The aim is to overcome these limitations and develop more robust skin cancer detection models.

4.1.1 Goals and objectives

Goal and Objectives:

- The project aims to develop a system that can accurately detect skin cancer at an early stage.
- The system will use deep learning techniques to analyze dermoscopic images.
- The system will be able to predict the specific type of skin cancer present in a given image irrespective of the noise in that image.

4.1.2 Statement of scope

The scope of this study encompasses the evaluation and improvement of skin cancer detection methods through the application of deep learning techniques. It involves an in-depth analysis of different convolutional neural networks (CNNs) and their performance in classifying skin lesions. The research will focus on addressing existing limitations, including low accuracy and the potential for overfitting when applying deep architectures like DenseNet-121. The scope also extends to the comparison of model performance on datasets, specifically the HAM10000 dataset and the ISIC dataset, to identify disparities in accuracy and dataset suitability. Additionally, the study aims to explore solutions for dataset selection and preparation concerns, ensuring that the research methods are transparent and reproducible. The project's scope is comprehensive, aiming to contribute to the advancement of accurate and robust skin cancer detection models.

4.2 SOFTWARE CONTEXT

The software context for this research project involves the development and implementation of deep learning software systems for the detection of skin cancer. This includes the utilization of various deep learning frameworks and libraries such as TensorFlow, PyTorch, and Keras. These frameworks enable the training and evaluation of convolutional neural networks (CNNs) for image classification. Data preprocessing, model training, and evaluation processes will be conducted using these software tools. Additionally, data management and analysis will be facilitated through Python libraries like Pandas and NumPy. Moreover, the software context encompasses the use of computer-aided design (CAD) software for the development of graphical user interfaces (GUIs) that may be employed for the visualization and interpretation of skin cancer detection results. This software will provide a user-friendly interface for clinicians and researchers to interact with and understand the model's outcomes.

4.3 MAJOR CONSTRAINTS

Constraints can be defined as limiting factor or state of restriction or lack of spontaneity of a software. Constraints are the limitations, hurdles which stop the software team from fulfilling their responsibility. Constraints are anything that restricts or dictates the actions of the project team. Any constraints that will impact the manner in which the software is to be specified, designed, implemented or tested are noted here.

3 core major constraints:

- Time - This refers to the actual time required to produce a deliverable which in this case would be the end result of the project. Naturally, the amount of time required to produce the deliverable will be directly related to the amount of requirements that are part of the end result along with the amount of resources allocated to the project.

- Cost - This is the estimation of the amount of money that will be required to complete the project. Cost itself encompasses various things, such as resources, labor rates for contractors, risk estimates, bills of materials, etc. All aspects of the project that have a monetary component are made part of the overall cost structure.

- Scope - These are the functional elements that, when completed, make up the end deliverable for the project. The scope itself is generally identified up front so as to give the project the best chance of success. Common success measure for the scope aspect of a project is its inherent quality upon delivery.

4.4 METHODOLOGIES OF PROBLEM SOLVING AND EFFICIENCY ISSUES

The methodologies employed for problem-solving and addressing efficiency issues within this project primarily revolve around the application of deep learning techniques in skin cancer detection. Through extensive literature review and the analysis of prior research, we aim to identify existing limitations and challenges in the field. By employing convolutional neural networks (CNNs) and various deep learning architectures, we intend to develop robust models capable of accurate skin cancer classification. Furthermore, we will assess the efficiency of these models, considering factors such as overfitting and dataset selection. Our approach involves comprehensive data collection, preprocessing, and model training to improve accuracy while mitigating overfitting risks. By comparing model performance on different datasets, particularly HAM10000 and ISIC, we seek to address dataset-related efficiency concerns. Through transparency in dataset selection and preparation, we aim to ensure reproducibility and reliability in our findings. The project's methodologies are designed to address both problem-solving and efficiency optimization to contribute meaningfully to the field of skin cancer detection.

Algorithm and Technique:

- Convolutional Neural Network
- K Nearest Neighbour
- Support Vector Machine

- Random Forest

4.5 SCENARIO IN WHICH MULTI-CORE, EMBEDDED AND DISTRIBUTED COMPUTING USED

Cloud Computing is a new but increasingly mature model of enterprise IT infrastructure that provides on-demand high quality applications and services from a shared pool of configuration computing resources .The cloud customers, individuals or enterprises, can outsource their local complex data system into the cloud to avoid the costs of building and maintaining a private storage infrastructure, as the cloud server possesses powerful functionality and flexibility. However, some problems may be caused in this circumstance since the Cloud Service Provider (CSP) possesses full control of the outsourced data. Cloud computing service models,

- Software as a service: It is a software delivery model in which applications are hosted by a service provider and make available to customers over a network.
- Platform as a service: It is a development platform provided as a service which supports the full software life cycle. It allows users to develop cloud based services and applications.
- Infrastructure as a service: It is a cloud service model that provides basic data storage and computing capabilities as standardized services over the network.

4.6 OUTCOME

The anticipated outcome of this project is the development of enhanced skin cancer detection models, addressing current limitations in accuracy and dataset selection. These models aim to contribute to more effective early detection of skin cancer through the application of advanced deep learning techniques.

4.7 APPLICATIONS

The applications of this research project on skin cancer detection using deep learning techniques are far-reaching and impactful. Beyond the academic realm, the outcomes of this study have the potential to significantly benefit the medical community and patients. Accurate and early detection of skin cancer through advanced algorithms can aid medical professionals in making more precise diagnoses and treatment decisions, ultimately improving patient outcomes. Additionally, the research findings can be leveraged for the development of automated skin cancer detection tools, which can be integrated into healthcare systems, telemedicine platforms, and smartphone applications, enhancing accessibility to healthcare services. Furthermore, the insights gained from this research can contribute to the ongoing efforts in the field of artificial intelligence-driven medical diagnostics, paving the way for the advancement of technology-driven healthcare solutions with broad implications for skin cancer detection and other medical applications.

4.8 HARDWARE RESOURCES REQUIRED

- Processor - Pentium IV/Intel I3 core
- Speed - 1.1 GHz
- RAM - 512 MB (min)
- Hard Disk - 20GB
- Keyboard - Standard Keyboard
- Mouse - Two or Three Button Mouse
- Monitor - LED Monitor

4.9 SOFTWARE RESOURCES REQUIRED

- Operating System: Windows 10 or macOS
- Programming Language: Python
- Python Libraries: OpenCV, Scikit-learn, PyTorch, TensorFlow
- Integrated Development Environment (IDE): Jupyter Notebook or Visual Studio Code

- Version Control: Git
- Dataset: Dermoscopic image dataset (e.g., HAM10000, ISIC)
- Other Resources: High-performance computing (HPC) cluster or GPU for deep learning model training and testing
- Documentation and Report Writing: LaTeX or Microsoft Word
- Reference Management: Zotero or Mendeley
- Visualization Tools: Matplotlib or Seaborn for data visualization
- Collaboration and Communication: Microsoft Teams or Slack for team communication
- Virtual Environment (Optional): Anaconda for managing Python packages and environments

CHAPTER 5

PROJECT PLAN

5.1 PROJECT ESTIMATES

Use Waterfall model and associated streams derived from assignments 1,2, 3, 4 and 5(Annex A and B) for estimation.

1. Requirement Gathering: Requirement gathering and plan for the initial part of the project was as follows: Understanding the problem definition. Understanding the reliable factors for data sharing systems. Gathering information about required software. Gathering information about required Hardware Resources Preparing preliminary design of overall work flows of project deciding the modules required for overall execution.

2. Analysis : Analysis of system collects systems requirement. Detailed plan of project and estimation of budget is studied in this stage. It is useful for the future development of project. Document containing system requirements is end product of this phase. Management and user requirements, alternative plans are further described by this document.

3. Design: Design focuses on high level design like, software i.e. Net beans for J2ME software are needed and their interaction with each other, then interface design to visualize how the project going to look like.

4. Implementation: In this phase code is generated from the designs.

5. Testing: System is tested in this phase. Individual modules are tested then whole system is tested. The system is tested to check inter related working of modules i.e. integration testing and that the system does what the user requires.

6. Maintenance: Maintenance is required for system. Because of some unexpected input values into the system the, changes in the system occurred.

5.1.1 Reconciled Estimates

5.1.1.1 Cost Estimate

5.1.1.2 Time Estimates

5.1.2 Project Resources

Project resources [People, Hardware, Software, Tools and other resources] based on Memory Sharing, IPC, and Concurrency derived using appendices to be referred.

5.2 RISK MANAGEMENT W.R.T. NP HARD ANALYSIS

This section discusses Project risks and the approach to managing them.

5.2.1 Risk Identification

For risks identification, review of scope document, requirements specifications and schedule is done. Answers to questionnaire revealed some risks. Each risk is categorized as per the categories mentioned in [1]. Please refer table 5.1 for all the risks. You can refereed following risk identification questionnaire.

1. Have top software and customer managers formally committed to support the project?
2. Are end-users enthusiastically committed to the project and the system/product to be built?
3. Are requirements fully understood by the software engineering team and its customers?
4. Have customers been involved fully in the definition of requirements?
5. Do end-users have realistic expectations?
6. Does the software engineering team have the right mix of skills?
7. Are project requirements stable?
8. Is the number of people on the project team adequate to do the job?

9. Do all customer/user constituencies agree on the importance of the project and on the requirements for the system/product to be built?

5.2.2 Risk Analysis

The risks for the Project can be analyzed within the constraints of time and quality

ID	Risk Description	Probability	Impact		
			Schedule	Quality	Overall
1	System Failure	Low	Low	High	High
2	Connection Failure	Low	Low	High	High

Table 5.1: Risk Table

Probability	Value	Description
High	Probability of occurrence is	$> 75\%$
Medium	Probability of occurrence is	$26 - 75\%$
Low	Probability of occurrence is	$< 25\%$

Table 5.2: Risk Probability definitions [1]

Impact	Value	Description
Very high	$> 10\%$	Schedule impact or Unacceptable quality
High	$5 - 10\%$	Schedule impact or Some parts of the project have low quality
Medium	$< 5\%$	Schedule impact or Barely noticeable degradation in quality Low Impact on schedule or Quality can be incorporated

Table 5.3: Risk Impact definitions [1]

5.2.3 Overview of Risk Mitigation, Monitoring, Management

Following are the details for each risk.

Risk ID	1
Risk Description	Description 1
Category	Development Environment.
Source	Software requirement Specification document.
Probability	Low
Impact	High
Response	Mitigate
Strategy	Strategy
Risk Status	Occurred

Risk ID	2
Risk Description	Description 2
Category	Requirements
Source	Software Design Specification documentation review.
Probability	Low
Impact	High
Response	Mitigate
Strategy	Better testing will resolve this issue.
Risk Status	Identified

Risk ID	3
Risk Description	Description 3
Category	Technology
Source	This was identified during early development and testing.
Probability	Low
Impact	Very High
Response	Accept
Strategy	Example Running Service Registry behind proxy balancer
Risk Status	Identified

5.3 PROJECT SCHEDULE

5.3.1 Project task set

Major Tasks in the Project stages are:

- Task 1:Requirement Analysis (Base Paper Explanation).
- Task 2:Project Specification (Paper Work).
- Task 3:Technology Study and Design.
- Task 4:Coding and Implementation (Module Development).

5.3.2 Task network

Individual tasks and subtasks have interdependencies based on their sequence. A task network is a graphic representation of the task flow for a project. Project tasks and their dependencies are noted.

5.4 TEAM ORGANIZATION

The manner in which staff is organized and the mechanisms for reporting are noted.

5.4.1 Team structure

The team structure for the project is identified. Roles are defined.

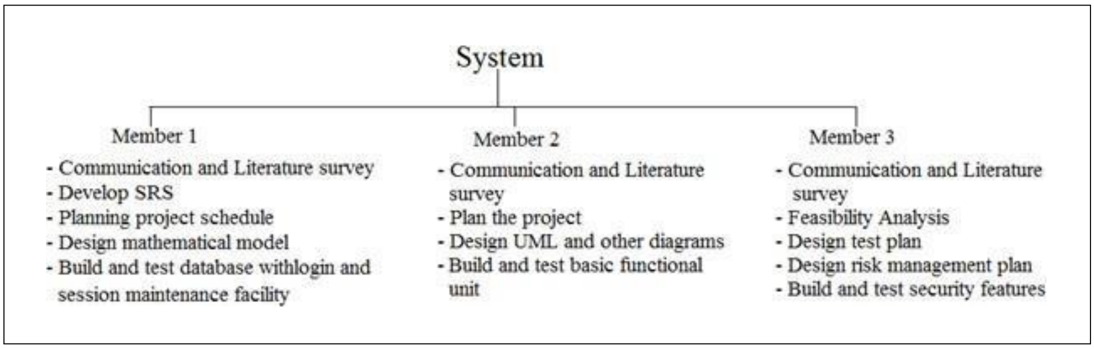

Figure 5.1: Team Structure

5.4.2 Management reporting and communication

Mechanisms for progress reporting and inter/intra team communication are identified as per assessment sheet and lab time table.

Sr No.	Month	Description
1	June	Discussion with guide regarding domain.
		Searching for IEEE paper for domain.
2	July	Short listing of IEEE papers within domain.
		Selection of IEEE paper.
3	August	Deciding Project name.
		Submission of Synopsis.
4	September	Requirement analysis.
		Designing of models.
5	October	Report preparation.
		Stage-I report submission.

Table 5.4: Management plan [1]

5.4.3 Timeline Chart

A project timeline chart is presented. This may include a time line for the entire project. Above points should be covered in Project Planner as Annex C and you can mention here Please refer Annex C for the planner

Task Name	Start date	End date	Duration
Initiate the project	04/8/2017	24/8/2017	21
Communication	04/8/2017	10/8/2017	7
Literature survey	11/8/2017	17/8/2017	7
Define scope	18/8/2017	19/8/2017	2
Develop SRS	20/8/2017	24/8/2017	5
Plan the project	25/8/2017	5/10/2017	42
Design model	25/8/2017	31/8/2017	7
Feasibility Analysis	01/9/2017	07/9/2017	7
Develop work breakdown structure	08/9/2017	09/9/2017	2
Planning project schedule	10/9/2017	14/9/2017	5
Design UML and other diagrams	15/9/2017	21/9/2017	7
Design test plan	22/9/2017	28/9/2017	7
Design risk management plan	29/9/2017	5/10/2017	7
Execute the project	05/01/2018	29/03/2018	84
Build and test basic model functionality	05/01/2018	25/01/2018	21
Build and test model accuracy	26/01/2018	15/02/2018	21
Build and test model interpretability	16/02/2018	08/03/2018	21
Build and test security features	09/03/2018	29/03/2018	21

Table 5.5: Time-line Chart [1]

CHAPTER 6

SOFTWARE REQUIREMENT SPECIFICATION

6.1 INTRODUCTION

6.1.1 Purpose and Scope of Document

1. Facilitating other Documentation: The SRS forms the basis for a load of other important documents such as the Software Design Specification.

2. Product Validation: It basically helps in validating with the client that the product which is being delivered, meets what they asked for.

6.1.2 Characteristics of a Software Requirement Specification

1. Accuracy: This is the first and foremost requirement. The development team will get nowhere if the SRS which will be the basis of the process of software development, is not accurate.

2. Completeness : The software requirement specification should not be missing any of the requirements stated in the business requirements documentation that the user specified.

3. Prioritization of Requirements : Software Requirement Specification should not simply be a wish list. The requirements should follow the order of priority and preference.

6.1.3 Overview of responsibilities of Developer

The responsibilities from developer perspective are to develop the application as per the requirement with user friendly user interface. For this the developer is required to go through the concepts required to be implemented. The developer follows the software design life cycle, starting from requirement gathering, analysis, design, coding, and then implementation of the system. After the implementation is completed, the graphical data representation and memory utilization is tested manually for random files or data.

6.2 USAGE SCENARIO

To provide the analyzing data from different perspectives and summarizing it into useful information - information that can be used to increase revenue, cuts costs, or both. It can be useful where huge amount of data is to be store. Ex. Google, Social Networking Application Etc.

6.2.1 User profiles

Sr No.	Actor	Description
1	User	The User has to firstly register itself by filling the registration form and be the active member of the system. If a User is already the member of the system then he or she can perform login process. Then Data Owner can be performing operations like file uploading, file downloading, sending request for encryption key, sending request for decryption key, sending request for OTP, etc.
2	Authority	The Authority has to firstly login himself. Then Authority can be performing operations like view encryption key request, view decryption key request, change password, etc.
3	Cloud Server	The Cloud Server has to firstly login himself. Then Cloud Server can be performing operations like view all users, view all files, view result, etc.

Figure 6.1: diagram

6.2.2 Use-cases

6.2.3 Use Case View

A use case diagram is a graphical representation of a user's interaction with the system and depicting the specifications of a use case. A use case diagram can show

Sr.No.	Use Case	Description	Actors	Assumptions
1	Register	The User have to firstly register itself by filling the registration form and be the active member of the system	User	Assumption
2	Login	User, Authority and Cloud Server has to login to perform operations	User, Authority, Cloud Server	Assumption
3	File Upload	User can be upload files on cloud server	User	Assumption
4	Request for encryption key	User can be send encryption key request for Authority	User	Assumption
5	View encryption key request	Authority can view all encryption key request	Authority	Assumption
6	Request for OTP	User can be send request for OTP to system for downloading other user's file	User	Assumption
7	Request for decryption key	User can be send request for decryption key to Authority for downloading other user's file	User	Assumption
8	View decryption key request	Authority van view all decryption key request	Authority	Assumption
9	Download file	User can be downloading file after entering OTP and decryption key	User	Assumption
10	View all users	Cloud server can be view all registered users	Cloud Server	Assumption
11	View all files	Cloud server can be view all uploaded file names	Cloud Server	Assumption
12	View result	Cloud server can be view our systems result	Cloud Server	Assumption
13	Logout	User, Authority, Cloud Server must logout after performing operations	User, Authority, Cloud Server	Assumption

Figure 6.2: diagram

the different types of users of a system and the various ways in which they interact with the system. Use case diagrams are used to gather the requirements of a system including internal and external influences. These requirements are mostly design requirements. So when a system is analyzed to gather its functionality use cases are prepared and actors are identified. The purposes of use case diagrams can be as follows:

- Used to gather requirements of a system.
- Used to get an outside view of a system.
- Identify external and internal factors influencing the system.
- Show the interaction among the actors.

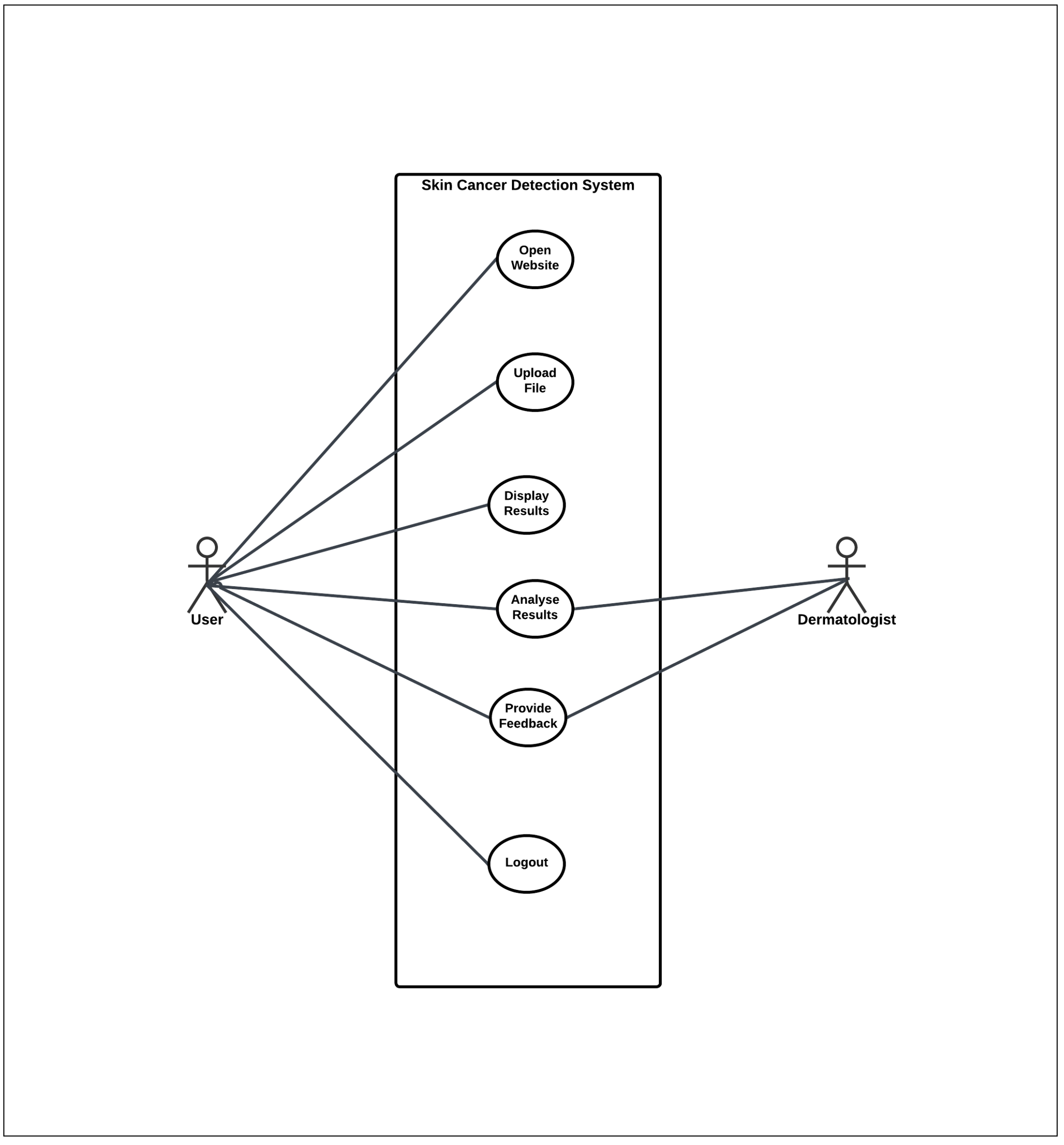

Figure 6.3: Use - Case diagram

6.3 DATA MODEL AND DESCRIPTION

6.3.1 Data Description

The controller is responsible for responding to user input and performs interactions on the data model objects. The controller receives the input; it validates the input and then performs the business operation that modifies the state of the data model.

6.3.2 Data objects and Relationships

The model is responsible for managing the data of the application. It responds to the request from the view and it also responds to instructions from the controller to update itself.

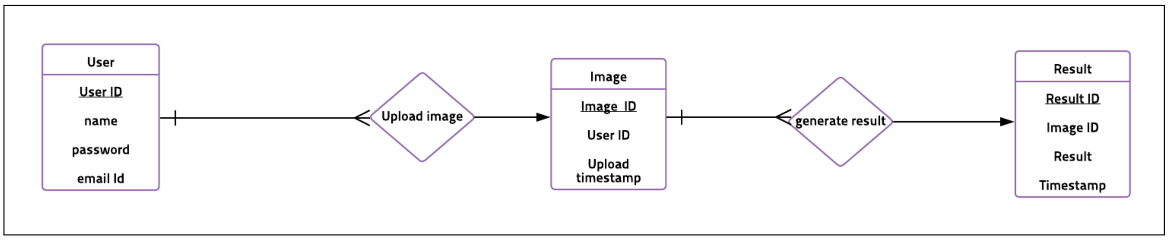

Figure 6.4: ER diagram

6.4 FUNCTIONAL MODEL AND DESCRIPTION

6.4.1 Data Flow Diagram

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modeling its process aspects. Often they are a preliminary step used to create an overview of the system which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design).A DFD shows what kinds of information will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about the timing of processes, or information about whether processes will operate in sequence or in parallel (which is shown on a flowchart).

6.4.1.1 Level 0 Data Flow Diagram

The DFD Level 0 identifies external entities and processes of the system. Level 0 explains the architecture that would be used for developing a software product. A context diagram is a top level (also known as ”Level 0”) data flow diagram. It only contains one process node (”Process 0”) that generalizes the function of the entire system in relationship to external entities.

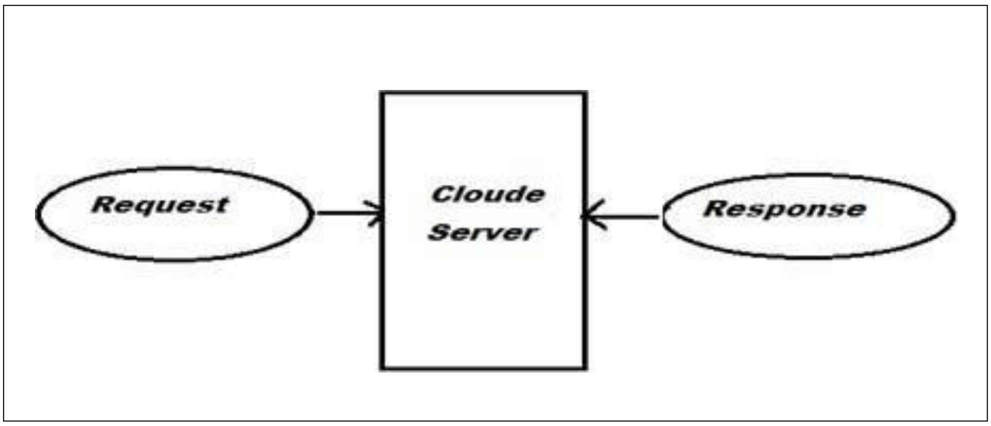

Figure 6.5: DFD0 diagram

6.4.1.2 Level 1 Data Flow Diagram

This DFD Level 1 shows the main processes in the work and the entities involved in it. We usually begin withdrawing a context diagram, a simple representation of the whole system. To elaborate further from that, we drill down to a level 1 diagram with additional information about the major functions of the system. Level 1 is an extension of level 0 diagram.

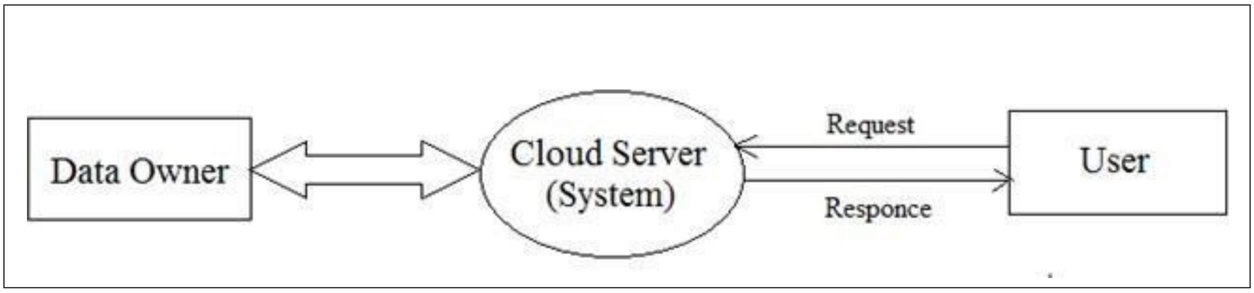

Figure 6.6: DFD1 diagram

6.4.2 Description of functions

A description of each software function is presented. A processing narrative for function n is presented.(Steps)/ Activity Diagrams. For Example Refer 6.7

6.4.3 Activity Diagram:

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams are intended to model both computational and organizational processes (i.e. workflows). Activity diagrams show the overall flow of control. Activity diagrams are constructed from a limited number of shapes, connected with arrows. The most important shape types:

- Rounded rectangles represent actions;
- Diamonds represent decisions;
- Bars represent the start (split) or end (join) of concurrent activities;
- A black circle represents the start (initial state) of the workflow;
- An encircled black circle represents the end (final state).

Arrows run from the start towards the end and represent the order in which activities happen. Hence they can be regarded as a form of flowchart. Typical flowchart techniques lack constructs for ex-pressing concurrency. However, the join and split symbols in activity diagrams only resolve this for simple cases; the meaning of the model is not clear when they are arbitrarily combined with decisions or loops.

- Activity Diagram for User

6.4.4 Non Functional Requirements:

- Interface Requirements
- Performance Requirements
- Software quality attributes such as availability [related to Reliability], modifiability [includes portability, reusability, scalability] , performance, security, testability and usability[includes self adaptability and user adaptability]

6.4.5 State Diagram:

State Transition Diagram

Fig.6.8 example shows the state transition diagram of Cloud SDK. The states are represented in ovals and state of system gets changed when certain events occur. The transitions from one state to the other are represented by arrows. The Figure shows important states and events that occur while creating new project.

6.4.6 Design Constraints

Any design constraints that will impact the subsystem are noted.

6.4.7 Software Interface Description

The software interface(s)to the outside world is(are) described. The requirements for interfaces to other devices/systems/networks/human are state.

CHAPTER 7

DETAILED DESIGN DOCUMENT USING APPENDIX A AND B

7.1 INTRODUCTION

This document specifies the design that is used to solve the problem of Product.

7.2 ARCHITECTURAL DESIGN

A description of the program architecture is presented. Subsystem design or Block diagram,Package Diagram,Deployment diagram with description is to be presented.

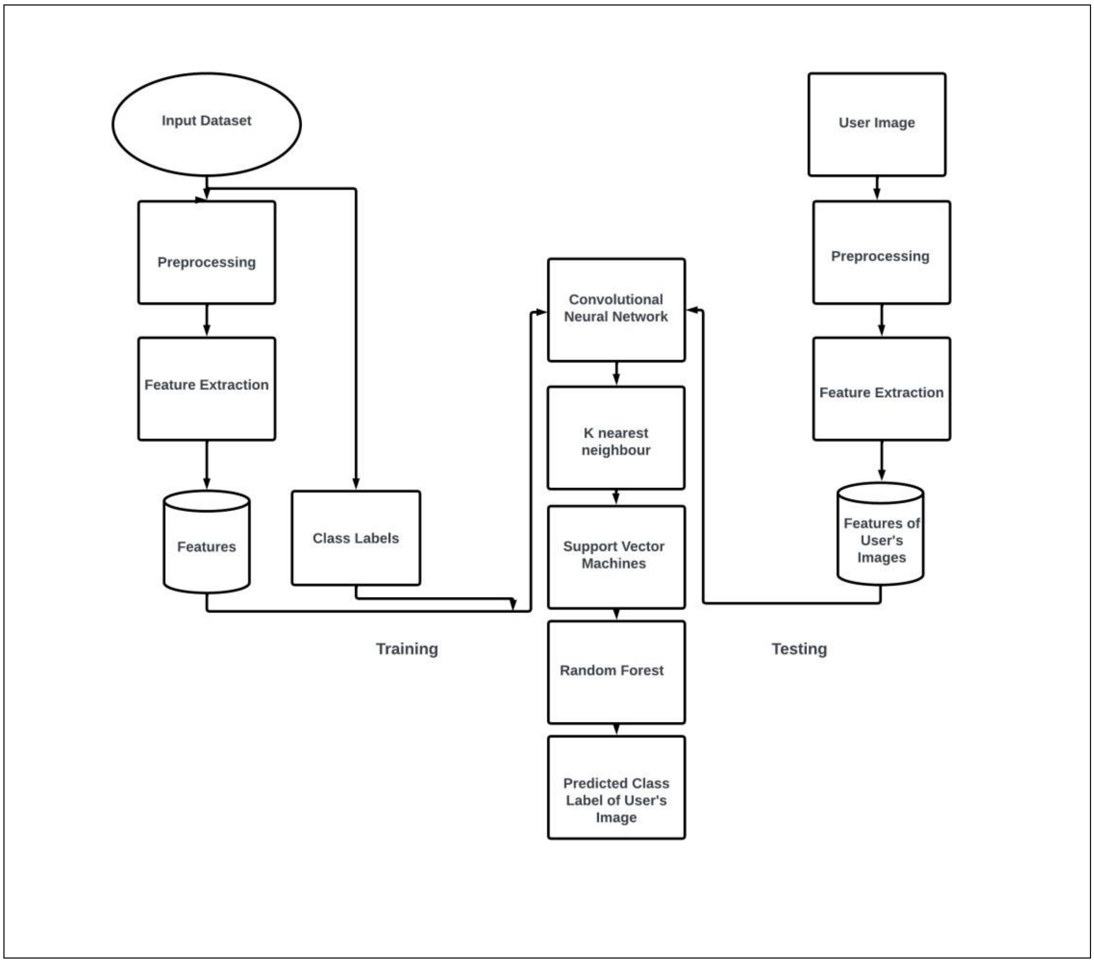

Figure 7.1: Architecture diagram

7.3 DATA DESIGN (USING APPENDICES A AND B)

A description of all data structures including internal, global, and temporary data structures, database design (tables), file formats.

7.3.1 Internal software data structure

Data structures that are passed among components the software are described.

7.3.2 Global data structure

Data structured that are available to major portions of the architecture are described.

7.3.3 Temporary data structure

Files created for interim use are described.

7.3.4 Database description

Database(s) / Files created/used as part of the application is(are) described.

7.4 COMPOENT DESIGN

Class diagrams, Interaction Diagrams, Algorithms. Description of each component description required.

7.4.1 Class Diagrams

CHAPTER 8

SUMMARY AND CONCLUSION

The primary objective of this project is to harness the capabilities of deep learning to develop a robust model that can accurately predict different types of skin cancer based on dermoscopic images. With a dataset comprising diverse images of skin cancer, including various types and subclasses, the model will undergo rigorous training, validation, and testing phases. The foundation of the project lies in the utilization of Convolutional Neural Networks (CNNs). These neural networks have gained renown for their exceptional image recognition capabilities and are well-suited to the task of identifying skin cancer from visual data. By leveraging CNNs and their ability to learn directly from data, the project aims to make a significant contribution to the field of dermatology and early cancer detection. The importance of early detection cannot be overstated, as skin cancer is not only prevalent but also highly treatable when diagnosed at an early stage. The project's focus on developing a model that can accurately differentiate between various types of skin cancer can be a game-changer in clinical practice. One of the unique aspects of this project is its holistic approach, covering everything from image acquisition to image segmentation and classification. Image acquisition is the pivotal initial step, involving the capture of high-quality images of skin lesions using techniques such as dermoscopy. This is followed by image segmentation, a crucial process where the pixels belonging to the lesion are separated from those of the surrounding skin. Once segmentation is achieved, the model can then classify the lesion, distinguishing between melanoma and non-melanoma cases. The technical aspects of the project are anchored in Python, utilizing libraries such as OpenCV, Scikit-learn, and PyTorch for image processing, machine learning, and deep learning. Furthermore, the project acknowledges the necessity of a diverse and extensive skin cancer image dataset for training and validation, ensuring the accuracy and robustness of the machine learning algorithms. Data privacy and security are also paramount, given the sensitive nature of medical data. The system is designed to uphold the privacy of patient information, ensuring that it complies with all ethical and legal standards.

In summary, this project holds the potential to significantly enhance healthcare by enabling the early and accurate detection of skin cancer.

CHAPTER 9

REFERENCES

Research article (Author/Year)	Objective/Proposed work	Methods/Techniques	Datasets	Relevant findings/Limitations identified
Inthiyaz et al.(2023)	Used pre-trained model for feature extraction and classification was performed using softmax classifier.	CNN	Xiangya-Derm	This work was tested on a very small dataset; these results can not be generalized on large datasets.Inthiyaz et al. achieved an AUC of 0.87, which can still be improved
Aljohani and Turki (2022)	Automatic Classification of Melanoma Skin Cancer with Deep Convolutional Neural Networks	DenseNet201, MobileNetV2, ResNet50V2, ResNet152V2, Xception, VGG16, VGG19, and GoogleNet	ISIC 2019	The maximum test accuracy of 76.09% was achieved using GoogleNet, which was quite low, and the model was tested only for the binary classification case.

Gajera et al.(2022)	Used eight pre-trained CNN models for the classification of dermoscopy images.	CNN model	ISIC 2016, and ISIC 2017 with only 200, 900, and 2000 training images.	Using deep architectures such as DenseNet-121 on small datasets may result in overfitting. Classification performance on the HAM10000 dataset was low.
Alenezi et al.（2022）	Multi-stage deep model was used for the extraction of features from skin cancer images.	Relief algorithm	ISIC-2019 and ISIC-2020）	the proposed work has limitations in terms of the time required for the parameter selection of the SVM classifier
Shinde et al. (2022）	Lightweight model was proposed for the classification of skin cancer images on IOT devices	***Squeeze-MNet Model***	ISIC	The proposed model in this work had lower sensitivity and specificity than other baseline models.
Bechelli and Delhommelle(2022)	Performance of different machine learning and deep learning algorithms was evaluated on skin cancer datasets.	CNN, pre-trained VGG-16, Xception, ResNet50	Kaggle dataset, HAM10000	The deep learning models did not perform well on the HAM10000 dataset, with the best-performing architecture ,accuracy was low than the accuracy achieved on the ISIC dataset which had fewer images.
Khan et al(2021）	Proposed CNN-based fully automated method for the classification and segmentation of images.	ResNet101, DenseNet201	Segmentation (ISBI 2016, ISBI 2017, ISIC 2018, PH2), classification (HAM10000)	The proposed model gave the best segmentation performance on PH2 , which has only 200 images; the effectiveness of the proposed method should be evaluated by testing it on larger datasets.
Adegun et al. (2021)	proposed a method for skin lesion segmentation based on a fully convolutional neural network.	fully convolutional neural network	ISBI 2017, PH2	HAM1000 dataset consists of seven classes; they did not mention how they selected a subset of HAM10000 or how they prepared the dataset, as the confusion matrix only has three classes.
Qasim Gilani et al.	Used spiking neural network (SNN) for the classification of skin cancer images.	Spiking VGG-13	HAM10000	The proposed spiking-VGG13 performed better, butthe specificity and precision of VGG-13 were higher.

[1] Inthiyaz et al.(2023) [1] ,Skin Cancer Detection using Deep Learning – A Review

[2] Aljohani and Turki (2022) [2] (3rd EdAutomatic Classification of Melanoma Skin Cancer with Deep Convolutional Neural Networks)

[3] Gajera et al.(2022)[3] Used eight pre-trained CNN models for the classification of dermoscopy images.

[4] Bechelli and Delhommelle(2022) [6], Performance of different machine learning and deep learning algorithms was evaluated on skin cancer datasets.

[5] Adegun et al. (2021) [8], “Big data: Proposed a method for skin lesion segmentation based on a fully convolutional neural network.

[6] Shinde et al. (2022)[5], Squeeze-MNet: Precise Skin Cancer Detection Model for Low Computing IoT Devices Using Transfer Learning

[7] Alenezi et al.(2022), Skin Cancer Detection – A Review

[8] Qasim Gilani et al.(2021) [9]Skin Cancer Detection using Deep Spiral Network.

[9] Khan et al(2021)[7], Proposed CNN-based fully automated method for the classification and segmentation of images.

ANNEXURE A

LABORATORY ASSIGNMENTS ON PROJECT ANALYSIS OF ALGORITHMIC DESIGN

- To develop the problem under consideration and justify feasibilty using concepts of knowledge canvas and IDEA Matrix.

  Refer [2] for IDEA Matrix and Knowledge canvas model. Case studies are given in this book. IDEA Matrix is represented in the following form. Knowledge canvas represents about identification of opportunity for product. Feasibility is represented w.r.t. business perspective.

I	D	E	A
Increase	Drive	Educate	Accelerate
Improve	Deliver	Evaluate	Associate
Ignore	Decrease	Eliminate	Avoid

Table A.1: IDEA Matrix

- Project problem statement feasibility assessment using NP-Hard, NP-Complete or satisfy ability issues using modern algebra and/or relevant mathematical models.

- input x,output y, y=f(x)

ANNEXURE B

LABORATORY ASSIGNMENTS ON PROJECT QUALITY AND RELIABILITY TESTING OF PROJECT DESIGN

It should include assignments such as

- Use of divide and conquer strategies to exploit distributed/parallel/concurrent processing of the above to identify object, morphisms, overloading in functions (if any), and functional relations and any other dependencies (as per requirements). It can include Venn diagram, state diagram, function relations, i/o relations; use this to derive objects, morphism, overloading
- Use of above to draw functional dependency graphs and relevant Software modeling methods, techniques including UML diagrams or other necessities using appropriate tools.
- Testing of project problem statement using generated test data (using mathematical models, GUI, Function testing principles, if any) selection and appropriate use of testing tools, testing of UML diagram's reliability. Write also test cases [Black box testing] for each identified functions. You can use Mathematica or equivalent open source tool for generating test data.
- Additional assignments by the guide. If project type as Entreprenaur, Refer [3],[4],[5], [6]

ANNEXURE C

PROJECT PLANNER

Using planner or alike project management tool.

ANNEXURE D

REVIEWERS COMMENTS OF PAPER SUBMITTED

(At-least one technical paper must be submitted in Term-I on the project design in the conferences/workshops in IITs, Central Universities or UoP Conferences or equivalent International Conferences Sponsored by IEEE/ACM)

1. Paper Title:
2. Name of the Conference/Journal where paper submitted :
3. Paper accepted/rejected :
4. Review comments by reviewer :
5. Corrective actions if any :

ANNEXURE E

PLAGIARISM REPORT

Plagiarism report