MINI PROJECT REPORT ON

MELANOMA SKIN CANCER DETECTION USING IMAGE PROCESSING AND DEEP LEARNING

Submitted in partial fulfillment of the requirement for the award of degree in

MASTER OF COMPUTER APPLICATIONS

of the

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

Submitted by

PRAVEENA K R (NCE21MCA-2035),

ASWATHY V S (NCE21MCA-2014),

DEVAKI V K (NCE21MCA-2018),

VISHNU A S (NCE21MCA-2052),

SUBITHA P (NCE21MCA-2049)

Under the guidance of
Mr. PRAMOD K

SENIOR ASSISTANT PROFESSOR

DEPARTMENT OF MCA

NEHRU COLLEGE OF ENGINEERING AND RESEARCH CENTRE,

(NAAC Re-Accredited with "A" grade) PAMPADY, THIRUVILWAMALA, THRISSUR - 680567

NOVEMBER 2022

MINI PROJECT REPORT ON

MELANOMA SKIN CANCER DETECTION USING IMAGE PROCESSING AND DEEP LEARNING

Submitted in partial fulfillment of the requirement for the award of degree in

MASTEROFCOMPUTERAPPLICATIONS

of the

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

Submitted by

PRAVEENA K R (NCE21MCA-2035),

ASWATHY V S (NCE21MCA-2014),

DEVAKI V K (NCE21MCA-2018),

VISHNU A S (NCE21MCA-2052),

SUBITHA P (NCE21MCA-2049)

Under the guidance of
Mr. PRAMOD K

SENIOR ASSISTANT PROFESSOR

DEPARTMENT OF MCA

NEHRU COLLEGE OF ENGINEERING AND RESEARCH CENTRE, NOVEMBER 2022

NEHRU COLLEGE OF ENGINEERING AND RESEARCH CENTRE

DEPARTMENT OF MCA

COLLEGE VISION

To mould true citizens who are millennium leaders and catalysts of change through excellence in education.

COLLEGE MISSION

NCERC is committed to transform itself into a center of excellence in Learning and Research in Engineering and Frontier Technology and to impart quality education to mould technically competent citizens with moral integrity, social commitment and ethical values. We intend to facilitate our students to assimilate the latest technological know-how and to imbibe discipline, culture and spiritually, and to mould them in to technological giants, dedicated research scientists and intellectual leaders of the country who can spread the beams of light and happiness among the poor and the underprivileged.

DEPARTMENT VISION

To create a school of distinction for the PG students, prepare them to be industry-ready, and achieve Academic excellence by continuous endorsement of the Faculty team in terms of Academics, Applications & Research.

DEPARTMENT MISSION

The Department of Computer Applications strives to provide quality and competency-based education and fine-tune the younger generation through Curricular, Co-Curricular and Extra-curricular activities so as to encounter the Professional and Personnel challenges ahead with Pragmatic skills & courage, thereby ‘Creating the True Citizens’.

NEHRU COLLEGE OF ENGINEERING AND RESEARCH CENTRE, PAMPADY

CERTIFICATE

This is to certify that, the work entitled **“MELANOMA SKIN CANCER DETECTION USING IMAGE PROCESSING AND DEEP LEARNING”** has been presented by **PRAVEENA K R (NCE21MCA-2035), ASWATHY V S (NCE21MCA-2014), DEVAKI V K (NCE21MCA-2018), VISHNU A S (NCE21MCA-2052), SUBITHA P (NCE21MCA-2049)** of Third Semester MCA in Partial Fulfillment of the requirement for the award degree **MASTER OF COMPUTER APPLICATIONS, APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY.**

We also certify that the work done is original.

Project guide **Head of the department**

Principal **External Examiner**

DECLARATION

We hereby declare that the project Report entitled "**MELANOMA SKIN CANCER DETECTION USING IMAGE PROCESSING AND DEEP LEARNING"** Submitted to the **Department of MCA at Nehru College of Engineering And Research Centre** in partial fulfillment of the requirement for the award of degree in **MASTER OF COMPUTER APPLICATIONS** from **APJ ABDUL KALAM TECHNLOGICAL UNIVERSITY,** is a record of original work done by me under the guidance of **Mr. PRAMOD K** Associate Professor of the Department of MCA, during my Third Semester MCA course period 2023.

PRAVEENA K R
ASWATHY V S
DEVAKI V K
VISHNU A S
SUBITHA P

ACKNOWLEDGEMNT

First and most, we thank the **God Almighty** for showing me the path to the completion of project work. I thank **Prof**. **Dr. Ambikadevi Amma T**, the principal **NCERC**, for providing good atmosphere for project completion and presentation. We thank **Prof**. **Dr. Sudheer S Marar**, Head of MCA Department, NCERC for his valuable suggestions and support throughout the project work. we wish to acknowledge our deep sense of gratitude and heartfelt thanks to my project guide **Mr. PRAMOD K** Senior Assistant Professor of the Department of MCA for their valuable suggestions, precious time, kind hearted motivations throughout the project work period. we express our immense gratitude to all my friends, without whom I would have never been able to do my project well.

ABSTRACT

Skin cancer is one of the most dangerous diseases in the world. Correctly classifying skin lesions at an early stage could aid clinical decision-making by providing an accurate disease diagnosis, potentially increasing the chances of cure before cancer spreads. However, achieving automatic skin cancer classification is difficult because the majority of skin disease images used for training are imbalanced and in short supply; meanwhile, the model's cross-domain adaptability and robustness are also critical challenges. Recently, many deep learning-based methods have been widely used in skin cancer classification to solve the above issues and achieve satisfactory results. Nonetheless, reviews that include the abovementioned frontier problems in skin cancer classification are still scarce. Therefore, in this article, we provide a comprehensive overview of the latest deep learning-based algorithms for skin cancer classification. We begin with an overview of three types of dermatological images, followed by a list of publicly available datasets relating to skin cancers. After that, we review the successful applications of typical convolutional neural networks for skin cancer classification. As a highlight of this paper, we next summarize several frontier problems, including data imbalance, data limitation, domain adaptation, model robustness, and model efficiency, followed by corresponding solutions in the skin cancer classification task. Finally, by summarizing different deep learning-based methods to solve the frontier challenges in skin cancer classification, we can conclude that the general development direction of these approaches is structured, lightweight, and multimodal. Besides, for readers' convenience, we have summarized our findings in figures and tables. Considering the growing popularity of deep learning, there are still many issues to overcome as well as chances to pursue in the future.

CONTENTS

List of Figures

List of Tables

Chapter 1

Introduction

Given the rising prevalence of skin cancer and the significance for early detection, it is crucial to develop an effective method to automatically classify skin cancer. As the largest organ of the human body, the skin shoulders the responsibility of protecting other human systems, which increases its vulnerability to disease. Melanoma was the most common cancer in both men and women with approximately 300,000 new cases diagnosed globally in 2018. In addition to melanoma, two other major skin cancer diseases, basal cell carcinoma (BCC) and squamous cell carcinoma (SCC), also had a relatively high incidence, with over 1 million cases in 2018. As reported, more skin cancers are diagnosed each year than all other cancers combined in the United States. Fortunately, if detected early, the chances of cure will be greatly improved. According to, melanoma has a 5-year survival rate of 99% when it does not metastasize. If it metastasizes to other organs in the body, its survival rate reduces to 20%. However, because early indications of skin cancer are not always visible, diagnostic results are often dependent on the dermatologist's expertise

. For inexperienced practitioners, an automatic diagnosis system is an essential tool for more accurate diagnoses. Beyond that, diagnosing skin cancer with naked eyes is highly subjective and rarely generalizable. Therefore, it is necessary to develop an automatic classification method for skin cancer that is more accurate, less expensive, and quicker to diagnose. Besides, implementing such automated diagnostic systems can effectively minimize mortality from skin cancers, benefiting both patients and the healthcare systems

.

However, owing to the complexity and diversity of skin disease images, achieving automatic classification of skin cancer is challenging. First of all, different skin lesions have lots of interclass similarities, which could result in misdiagnosis. For example, there exist various mimics of BCC in histopathological images, such as SCC and other skin diseases .As a result, it is difficult for the diagnosis systems to effectively discriminate skin malignancies from their known imitators. Secondly, several skin lesions differ within their same class in terms of color, feature, structure, size, and location. For example, the appearance of BCC and its subcategories is almost different. This makes it difficult to classify different subcategories of the same category. Furthermore, the classification algorithms are highly sensitive to the types of camera devices used to capture images. When the test images come from a different domain, their performance suffers.

Although traditional machine learning approaches are capable of performing well in particular skin cancer classification tasks, these algorithms are ineffective for complicated diagnostic demands in clinical practice. Traditional machine learning methods for skin cancer diagnosis typically involve extracting features from skin-disease images and then classifying the extracted features . For example, ABCD Rule ,Menzies Method , and 7-Point Checklist are effective methods for extracting various features from skin disease images. The handcrafted features are then classified using several classification methods such as SVM , XGBoost , and decision tree (. Due to the restricted number of selected features, machine learning algorithms can often only classify a subset of skin cancer diseases and cannot generalize to a broader range of disease types . Besides, given the wide variety of skin cancers, it is not effective to identify each form of cancer solely based on handcrafted features .

Without the need for domain expertise and feature extraction, deep learning algorithms have been widely used for skin cancer classification in recent years; however, there are still several difficulties and challenges ahead. Compared with traditional machine learning methods, deep learning algorithms can analyze data from a large-scale dataset faster and more accurately, which allows them to effectively extract relevant characteristics . At the same time, deep learning algorithms can also aid clinicians in more thorough data analysis and examination of test results .A number of studies, such as demonstrated that deep learning algorithms can diagnose at a level comparable to that of a dermatologist. However, these algorithms still have many obstacles to becoming a complete diagnostic system. Firstly, data imbalance and the lack of a large volume of labeled images have hindered the widespread use of deep learning methods in skin cancer classification . When these algorithms are used to classify skin cancers that are rare in the training dataset, they frequently result in a misdiagnosis . Furthermore, when working with high-resolution images (such as pathological images) with millions of pixels, the deep learning models often result in significant computing costs and additional training time . Besides, different noises will be generated as a result of the various conditions (such as different imaging devices, backgrounds). Therefore, the robustness and generalization ability of these algorithms should also be taken into account .

These years, a number of reviews that detail the diagnostic breakthroughs in skin cancer classification have been published; however, no review has provided a specific analysis of frontier challenges in skin cancer classification tasks, such as data imbalance and limitation, domain adaptability, model robustness, and model efficiency. reviewed the recent developments in skin lesion classification using dermoscopic images . presented a detailed overview of studies on using CNNs to classify skin lesions . showed how the use of CNNs in correctly identifying skin cancer has developed. presented a review of different machine learning algorithms in dermatology diagnosis, as well as some of the obstacles and limitations . and summarized a number of deep learning-based approaches for skin cancer classification, as well as various challenges and difficulties .provided an in-depth review of the current articles about melanoma classification and compared their results with human experts. summarized the latest CNN-based methods in skin lesion classification by utilizing image data and patient data . provided a review of deep learning-based methods for early diagnosis of skin cancer. We present these relevant surveys with details and highlights in Table 1. By summarizing the previous reviews, we find that all of the preceding publications methodically studied a specific topic in skin cancer classification. However, most of them treated skin cancer classification as a classical classification problem, without addressing the model's significant practical constraints in clinical work, such as data imbalance and limitation, cross-domain adaptability, model robustness, and model efficiency. Although several earlier reviews summarized some of the methods to solve the abovementioned frontier problems, their summaries were incomplete. Some novel techniques were not covered, such as pruning, knowledge distillation, and transformer. Therefore, in this review, we comprehensively summarize the frontier challenges in skin cancer classification and provide corresponding solutions by analyzing articles published until the year 2022. It gives readers in-depth information on the advances and limitations of deep learning in skin cancer classification and also provides different ideas for researchers to improve these algorithms.

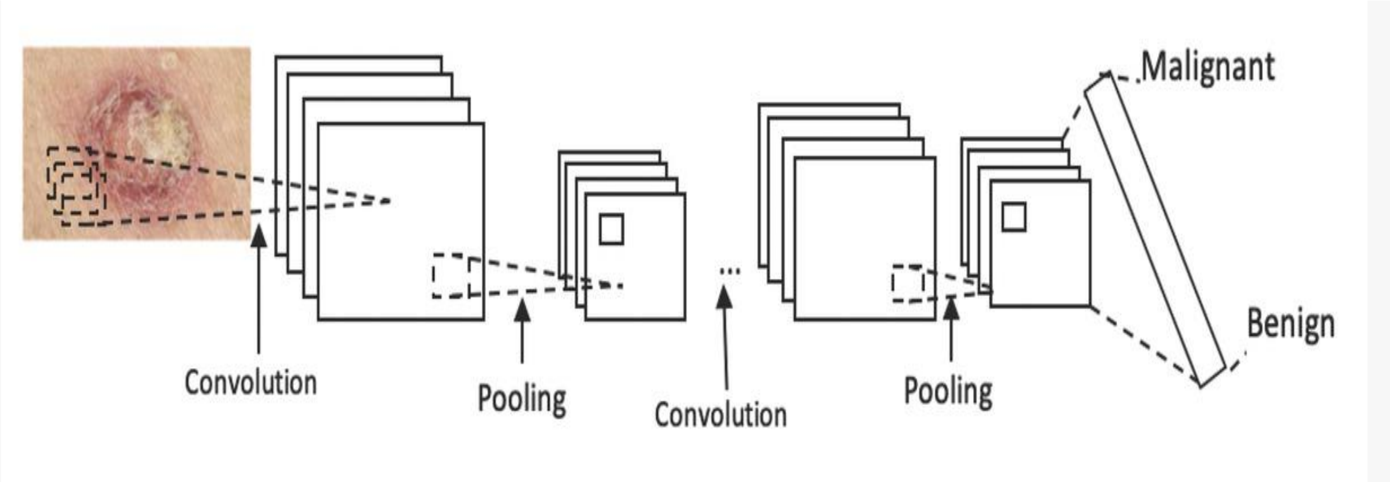


Fig1: Convolution Neural network for Skin Cancer detection

NEURAL NETWORK FOR MEDICAL ANALYSIS Computer vision is a way to understand or simulate human vision of things (objects, events, motion, etc.) and attempt to make computer to do the same, by simulating brain images and videos processing that come from the retina. Computer vision can be defined as a branch of Artificial Intelligence (AI) that focus on using geometry to transform digital image or sequence of images (video) into high dimensional data, and then extract features, such as texture, points, lines, edges, shapes, motions, and ridges. A common application of computer vision is medical image processing, which aims to obtain significant information from images to automate patient’s diagnosis. An example of this automation is computer-aided detection of breast cancer and skin cancer [7]. Artificial Neural Network (ANN) is the very fundamental of deep learning. ANN originally presented by the neuro physiologist Warren McCulloch and the mathematician Walter Pittsin 1943. They proposed the first neural network architecture (computational model) inspired from human biological neuron (Fig. 3) of how biological neurons are connected together in animals’ brains to transmit data and accomplish complex computations tasks. In 1960, further ANN architectures contributed to believe in machine intelligent,then ANN entered forgetting era till 1980 when gained interest again, but 1990 shown newly invented alternative machine learning techniques much powerful than ANN such as SVM (Support Vector Machines). Recently, ANN gained another life chance due to the huge amount of quality data and significant increase in computers storage and power which assisted ANN to produce better results. Nowadays, ANN and its related techniques are considered as a powerful machine learning tool to tackle large and highly complex tasks, for example, Google uses deep learning to classify tremendous number of images, Apple's Siri speech recognition tool, and YouTube for suggesting best related videos for the big number of users every day [5], [6]. Computer vision and image classification problem were among the main factors of development of deep learning, because AI researchers aimed to learn features from raw of images which require higher computations and memory resources in compare to text and numerical learning models.In particular, the network called AlexNet has been used and modified to solve computer vision and image classification problems which draw the map for the Convolutional Neural Network (CNN) .

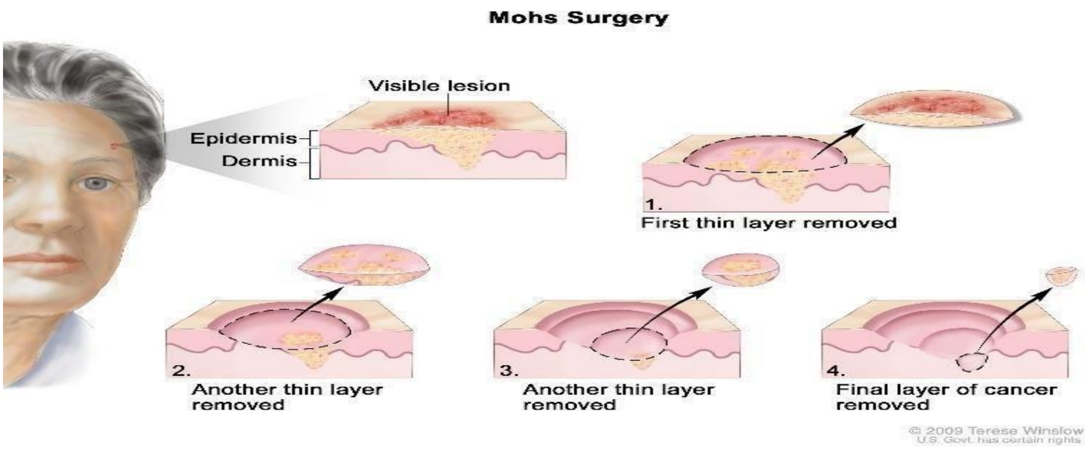


Fig2:

A Surgical Procedure Steps to Remove Skin Cancer

The increased popularity in Convolutional Neural Networks in medical analysis and computer vision is due to its outstanding performance in analyzing and classifying images. Consequently, CNN became one of the most popular models in deep learning and computer vision. The key idea behind convolutional neural networks is to build partially connected layers. For example, an image with shape 100 × 100 which form 10,000 pixels as input to the network and suppose the first layer consist of only 1000 neurons then the number of connections between input layer and first hidden layer will be about 10 million connections, which require huge computations and memory. However, CNN can resolve this issue using partially connected layers. In CNNs, there are receptive fields to connect the input layer to a feature map. Receptive fields can be defined as overlapping windows that travel over the pixels of an input image to create a feature map. The shifting length in input image window and the size of window itself is determined during model design and implementation. The process of creating the feature map is also called convolution. An example of such a convolutional layer, the layer that connects the input pixels to each unit in the feature map, is shown in Figure.

The convolutional neural network is typically consisting of three layers, convolution layer, pooling layer, and fully connected layer. In addition, CNN may contain optional layers such as dropout layer, however, convolution, pooling, and fully connected layers are the most popular architecture for CNN.

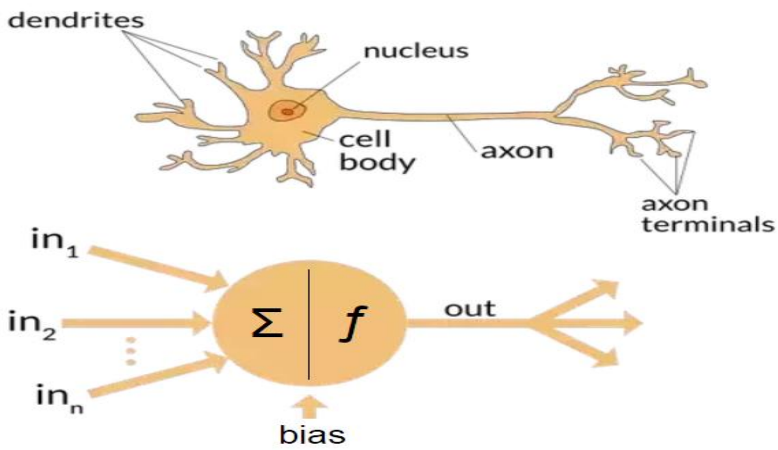
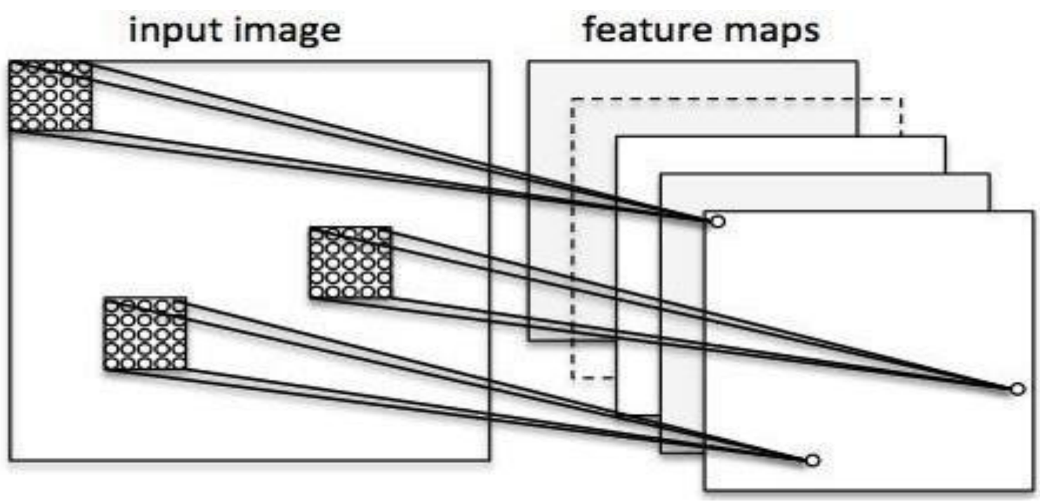


Fig3.



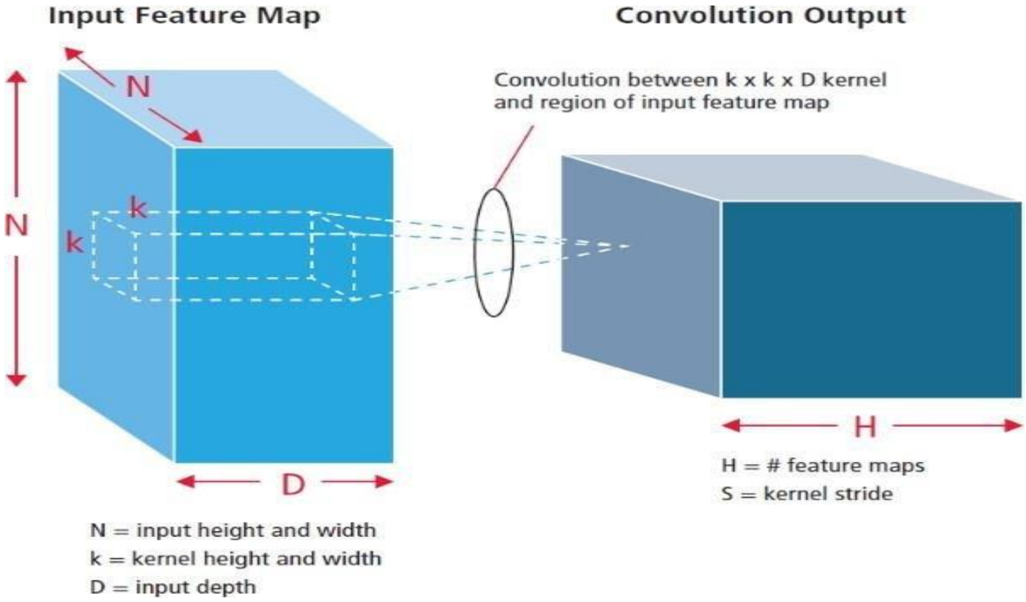
An Example of a Convolutional Layer, the Layer that Connects the Input Pixels to Each unit in the Feature Map

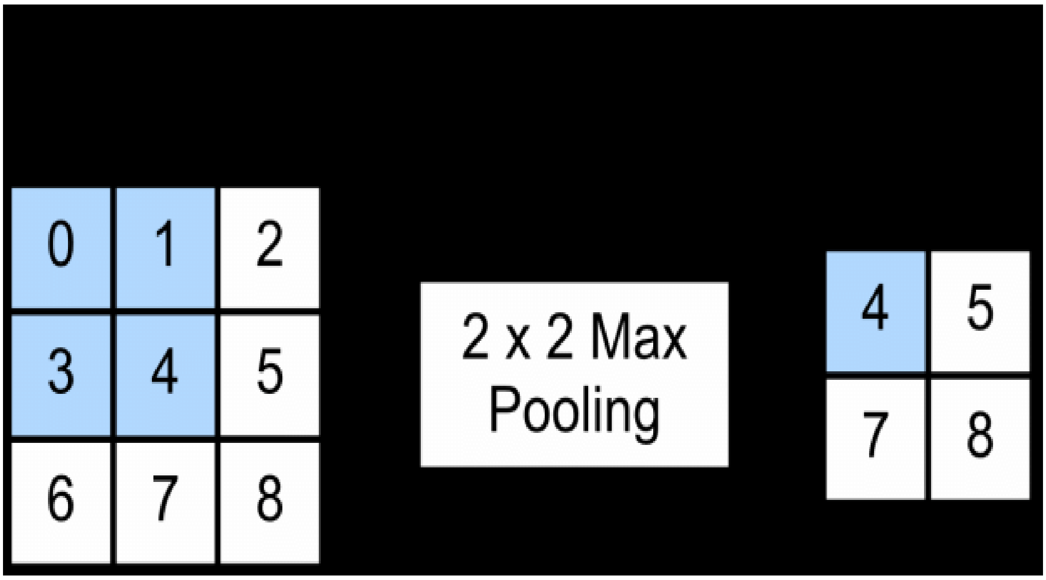
A. Convolution Layer

The convolution layer is the central structure for CNN and performs the most computational load. This layer performs a dot product between two matrices, the first matrix is the set filters also known as kernels and usually smaller in size than the image itself, for example, kernels for detecting vertical or horizontal edges. The second matrix is window matrix of the receptive field which is typically a portion of image with predefined size and sliding. Most of the images are composed of three channels (RGB), therefore, the kernel and receptive field will be also extending to three channels too. Initially, the kernel travel across input image with to produce smaller image that map input image with kernel, sample output of convolutional layer is shown in Figure.

B. Pooling Layer

The main purpose of pooling layer is to reduces the size of output which produced from previous layer (convolution layer), reducing the size of features will reduce the computation time and makes features robust against noise and outliers. There are several ways to do pooling such as max pooling and average pooling, however, the most popular is max pooling which calculate the maximum output from the neighborhood as shown in Fig. 6 [12]. Sample fully connected layer is shown in Figure.


Input Feature Map
Convolution Output
N
N
k
k
Convolution between k x k x D kernel
and region of input feature map
D
H
N = input height and width
k = kernel height and width
D = input depth
H = # feature maps
S = kernel stride


0
1
2
3
4
5
6
7
8
2 x 2 Max Pooling
4
5
7
8

1.1 Background

Datasets

To create a trustworthy and robust skin cancer classification system, a variety of datasets with all kinds of dermatological images are required. As the need for medical imaging resources in academia grows, more and more datasets are becoming publicly available. To provide readers with a reference, we introduce several commonly used skin-disease datasets in the next part, along with the works based on these datasets

1.2 Motivation

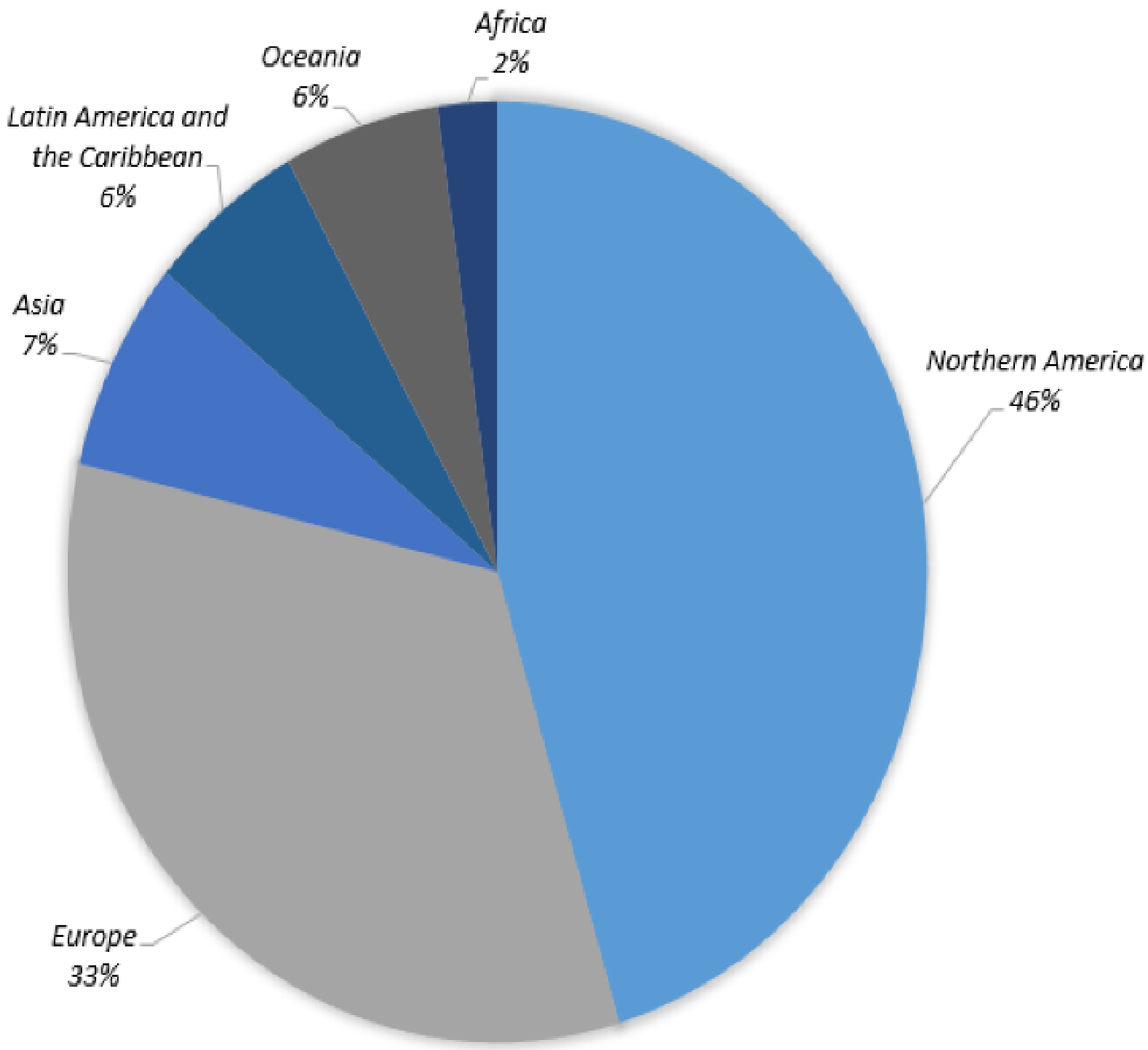
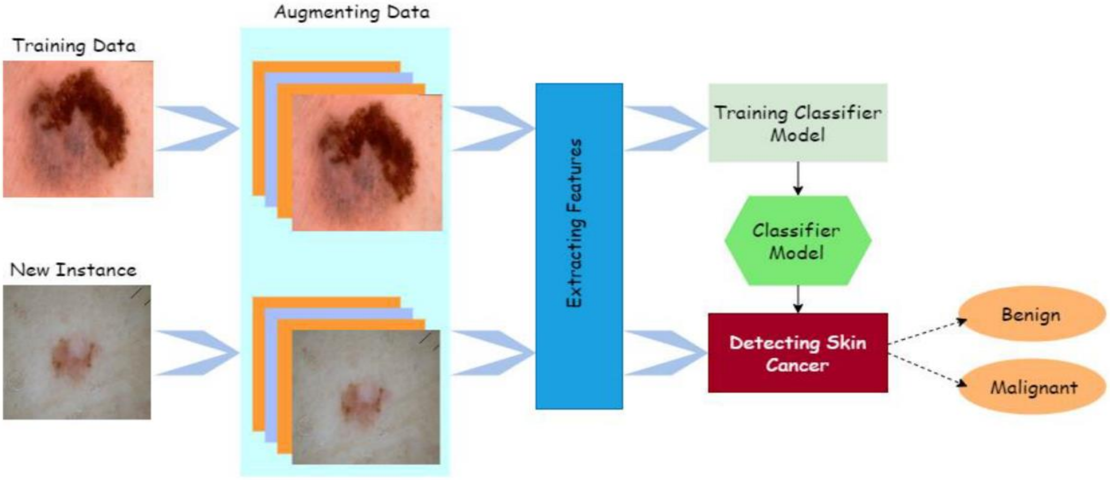


Fig4.

Skin cancer cases globally (22 March 2022)



The uncontrollable development of tissues in a specific body area is known as cancer . One of the most quickly spreading diseases in the world looks to be skin cancer. Skin cancer is a disease in which abnormal skin cells develop out of control. To ensure better prognosis and death rates, early skin cancer identification is crucial, yet solid tumor detection typically relies mostly on screening mammography with inadequate sensitivity, which is then validated by clinical specimens. Cancer screening and treatment reaction evaluations are usually not appropriate uses for this approach [2,3]. An increasing number of healthcare providers are using artificial intelligence (AI) for medical diagnostics to improve and accelerate the diagnosis decision-making procedure [4]. However, despite some current evidence of improvement in this domain, the accurate assessment and adequate reporting of predicted flaws have been entirely or partly ignored by currently available AI research for clinical diagnosis

1.3.Objective

fig5. Process of cancer detection

The exponential growth in processing power has led to tremendous advancements in computer vision technologies, particularly in the development of deep learning models such as CNN. The earliest possible detection of skin cancer is now required. Skin cancer is the second most common cancer (after breast cancer) in women between the ages of 30 and 35, and the most common cancer in women between the ages of 25 and 29, according to Dr. Lee , who serves several young patients with skin cancer. Early identification of skin cancer using deep learning outperformed human specialists in many computer vision challenges, resulting in reduced death rates. It is possible to get outstanding and cutting-edge processing and classification accuracy by including efficient formulations into deep learning techniques.

In order to correctly diagnose early cancer signs from lesion images, this study proposes a crossbred DL model for cancer classification and prediction. Preprocessing and classification are key components of the system under consideration. During the preprocessing phase, the entire intensity of the image is improved to decrease the inconsistencies among photos. The image is additionally scaled and standardized to fit the training model's scale during this procedure. Many different metrics were used to evaluate the suggested model in the comparison studies. These metrics included precision and recall metrics, the F1-score, and the area under the curve (AUC). The publicly available, large-scale ISIC 2018 dataset comprises a massive number of lesion images with diagnosed cancer. Pretrained networks such as Resnet50, InceptionV3, and Inception Resnet were employed for comparison. A training process with varying configurations of training strategies (e.g., validation patience and data augmentation) was employed to boost the recommended technique's universal efficiency and prevent overfitting.

1.3 Contribution

The major contributions in this project are:

- More user friendly
- Easiness of cancer detection
- Viewing the result accurately
- Implementation of Advanced web Technology

1.4 Report Organization

The project report is divided into six sections. Section 2 describes literature survey. Section 3 describes the methodology and section 4 describes agile methodology used for implementing the project. Section 5 gives the results and discussions. Finally, Section 6 gives the conclusion.

Chapter 2

Literature Survey

Paper 1:

In the diagnosis of skin melanoma by analyzing histopathological images, the detection of the melanocytes in the epidermis area is an important step. However, the detection of melanocytes in the epidermis area is difficult because other keratinocytes that are very similar to the melanocytes are also present. This paper [1] proposes a novel computer-aided technique for segmentation of the melanocytes in the skin histopathological images. In order to reduce the local intensity variant, a mean-shift algorithm is applied for the initial segmentation of the image. A local region recursive segmentation algorithm is then proposed to filter out the candidate nuclei regions based on the domain prior knowledge. To distinguish the melanocytes from other keratinocytes in the epidermis area, a novel descriptor, named local double ellipse descriptor (LDED), is proposed to measure the local features of the candidate regions. The LDED uses two parameters: region ellipticity and local pattern characteristics to distinguish the melanocytes from the candidate nuclei regions. Experimental results on 28 different histopathological images of skin tissue with different zooming factors show that the proposed technique provides a superior performance.

Paper 2:

Melanoma is the deadliest form of skin cancer. Incidence rates of melanoma have been increasing, especially among non-Hispanic white males and females, but survival rates are high if detected early. Due to the costs for dermatologists to screen every patient, there is a need for an automated system to assess a patient's risk of melanoma using images of their skin lesions captured using a standard digital camera [2]. One challenge in implementing such a system is locating the skin lesion in the digital image. A novel texture-based skin lesion segmentation algorithm is proposed. A set of representative texture distributions are learned from an illumination-corrected photograph and texture distinctiveness metric is calculated for each distribution. Next, regions in the image are classified as normal skin or lesion based on the occurrence of representative texture distributions. The proposed segmentation framework is tested by comparing lesion segmentation results and melanoma classification results to results using other state-of-art algorithms. The proposed framework has higher segmentation accuracy compared to all other tested algorithms.

Paper 3:

Melanoma spreads through metastasis, and therefore, it has been proved to be very fatal. Statistical evidence has revealed that the majority of deaths resulting from skin cancer are as a result of melanoma. Further investigations have shown that the survival rates in patients depend on the stage of the cancer; early detection and intervention of melanoma implicate higher chances of cure. Clinical diagnosis and prognosis of melanoma are challenging, since the processes are prone to misdiagnosis and inaccuracies due to doctors' subjectivity. Malignant melanomas are asymmetrical, have irregular borders, notched edges, and color variations, so analyzing the shape, color, and texture of the skin lesion is important for the early detection and prevention of melanoma. This paper proposes the two major components of a noninvasive real-time automated skin lesion analysis system for the early detection and prevention of melanoma. The first component is a real-time alert to help users prevent skinburn caused by sunlight; a novel equation to compute the time for skin to burn is thereby introduced. The second component is an automated image analysis module, which contains image acquisition, hair detection and exclusion, lesion segmentation, feature extraction, and classification. The proposed system [3] uses PH2 Dermoscopy image database from Pedro Hispano Hospital for the development and testing purposes. The image database contains a total of 200 dermoscopy images of lesions, including benign, atypical, and melanoma cases. The experimental results show that the proposed system is efficient, achieving classification of the benign, atypical, and melanoma images with accuracy of 96.3%, 95.7%, and 97.5%, respectively.

Paper 4:

Several efforts have been engaged in the development of computer systems that aid in the diagnosis of skin lesions and in particular melanoma. In the present paper [4] presents the development of a dermoscopic image classification system for mobile devices. The system analyzes the skin lesions from the edge characteristics and color of the ABCD rule, and classifies the lesion using a MLP network trained with the backpropagation algorithm. Whose goal is to provide a tool capable of conducting the examination of dermoscopy in any location and allow it to be carried out by different health professionals. According to the results, the system achieved a level of 66% sensitivity and 93% specificity.

Paper 5:

Segmentation of the wrist bones in CT images has been frequently used in different clinical applications including arthritis evaluation, bone age assessment and image-guided interventions. The major challenges include non-uniformity and spongy textures of the bone tissue as well as narrow inter-bone spaces. In this work [5], we propose an automatic wrist bone segmentation technique for CT images based on a statistical model that captures the shape and pose variations of the wrist joint across 60 example wrists at nine different wrist positions. To establish the correspondences across the training shapes at neutral positions, the wrist bone surfaces are jointly aligned using a group-wise registration framework based on a Gaussian Mixture Model. Principal component analysis is then used to determine the major modes of shape variations. The variations in poses not only across the population but also across different wrist positions are incorporated in two pose models. An intra-subject pose model is developed by utilizing the similarity transforms at all wrist positions across the population. Further, an inter-subject pose model is used to model the pose variations across different wrist positions. For segmentation of the wrist bones in CT images, the developed model is registered to the edge point cloud extracted from the CT volume through an expectation maximization based probabilistic approach. Residual registration errors are corrected by application of a non-rigid registration technique. We validate the proposed segmentation method by registering the wrist model to a total of 66 unseen CT volumes of average voxel size of 0.38 mm. We report a mean surface distance error of 0.33 mm and a mean Jaccard index of 0.86.

Paper 6:

The ABCD (asymmetry, border irregularity, colour and dermoscopic structure) rule of dermoscopy is a scoring method used by dermatologists to quantify dermoscopy findings and effectively separate melanoma from benign lesions. Automatic detection of the ABCD features and separation of benign lesions from melanoma could enable earlier detection of melanoma. In this study [6], automatic ABCD scoring of dermoscopy lesions is implemented. Pre-processing enables automatic detection of hair using Gabor filters and lesion boundaries using geodesic active contours. Algorithms are implemented to extract the characteristics of ABCD attributes. Methods used here combine existing methods with novel methods to detect colour asymmetry and dermoscopic structures. To classify lesions as melanoma or benign nevus, the total dermoscopy score is calculated. The experimental results, using 200 dermoscopic images, where 80 are malignant melanomas and 120 benign lesions, show that the algorithm achieves 91.25% sensitivity of 91.25 and 95.83% specificity. This is comparable to the 92.8% sensitivity and 90.3% specificity reported for human implementation of the ABCD rule. The experimental results show that the extracted features can be used to build a promising classifier for melanoma detection.

Paper 7:

Automatic skin lesion segmentation in dermoscopic images is a challenging task due to the low contrast between lesion and the surrounding skin, the irregular and fuzzy lesion borders, the existence of various artifacts, and various imaging acquisition conditions. In this paper [7], a fully automatic method for skin lesion segmentation by leveraging 19-layer deep convolutional neural networks that is trained end-to-end and does not rely on prior knowledge of the data. We propose a set of strategies to ensure effective and efficient learning with limited training data. Furthermore, we design a novel loss function based on Jaccard distance to eliminate the need of sample re-weighting, a typical procedure when using cross entropy as the loss function for image segmentation due to the strong imbalance between the number of foreground and background pixels. We evaluated the effectiveness, efficiency, as well as the generalization capability of the proposed framework on two publicly available databases. One is from ISBI 2016 skin lesion analysis towards melanoma detection challenge, and the other is the PH2 database. Experimental results showed that the proposed method outperformed other state-of-the-art algorithms on these two databases. This method is general enough and only needs minimum pre- and post-processing, which allows its adoption in a variety of medical image segmentation tasks.

Paper 8:

The segmentation of skin lesions in dermoscopic images is a fundamental step in automated computer-aided diagnosis of melanoma. Conventional segmentation methods, however, have difficulties when the lesion borders are indistinct and when contrast between the lesion and the surrounding skin is low. They also perform poorly when there is a heterogeneous background or a lesion that touches the image boundaries; this then results in under and over segmentation of the skin lesion [8]. Here suggested that saliency detection using the reconstruction errors derived from a sparse representation model coupled with a novel background detection can more accurately discriminate the lesion from surrounding regions. Further proposed a Bayesian framework that better delineates the shape and boundaries of the lesion. Here it is also evaluated the approach on two public datasets comprising 1100 dermoscopic images and compared it to other conventional and state-of-the-art unsupervised (i.e., no training required) lesion segmentation methods, as well as the state-of-the-art unsupervised saliency detection methods. The results show that this approach is more accurate and robust in segmenting lesions compared to other methods.

Paper 9:

Segmentation of skin lesions is an important step in the automated computer aided diagnosis of melanoma. However, existing segmentation methods have a tendency to over- or under-segment the lesions and perform poorly when the lesions have fuzzy boundaries, low contrast with the background, inhomogeneous textures, or contain artifacts. Furthermore, the performance of these methods are heavily reliant on the appropriate tuning of a large number of parameters as well as the use of effective preprocessing techniques, such as illumination correction and hair removal. Proposed [9] to leverage fully convolutional networks (FCNs) to automatically segment the skin lesions. FCNs are a neural network architecture that achieves object detection by hierarchically combining low-level appearance information with high-level semantic information. Addressed the issue of FCN producing coarse segmentation boundaries for challenging skin lesions (e.g., those with fuzzy boundaries and/or low difference in the textures between the foreground and the background) through a multistage segmentation approach in which multiple FCNs learn complementary visual characteristics of different skin lesions; early stage FCNs learn coarse appearance and localization information while late-stage FCNs learn the subtle characteristics of the lesion boundaries. Introduced a new parallel integration method to combine the complementary information derived from individual segmentation stages to achieve a final segmentation result that has accurate localization and well-defined lesion boundaries, even for the most challenging skin lesions. In results, achieved an average Dice coefficient of 91.18% on the ISBI 2016 Skin Lesion Challenge dataset and 90.66% on the PH2 dataset. Extensive experimental results on two well-established public benchmark datasets demonstrate that this method is more effective than other state-of-the-art methods for skin lesion segmentation.

Paper 10:

Automated melanoma recognition in dermoscopy images is a very challenging task due to the low contrast of skin lesions, the huge intraclass variation of melanomas, the high degree of visual similarity between melanoma and non-melanoma lesions, and the existence of many artifacts in the image. In order to meet these challenges, proposed [10], a novel method for melanoma recognition by leveraging very deep convolutional neural networks (CNNs). Compared with existing methods employing either low-level hand-crafted features or CNNs with shallower architectures, our substantially deeper networks (more than 50 layers) can acquire richer and more discriminative features for more accurate recognition. To take full advantage of very deep networks, proposed a set of schemes to ensure effective training and learning under limited training data. First, the residual learning to cope with the degradation and over fitting problems is applied when a network goes deeper. This technique can ensure that our networks benefit from the performance gains achieved by increasing network depth. Then, we construct a fully convolutional residual network (FCRN) for accurate skin lesion segmentation, and further enhance its capability by incorporating a multi-scale contextual information integration scheme. Finally, seamlessly integrate the proposed FCRN (for segmentation) and other very deep residual networks (for classification) to form a two-stage framework. This framework enables the classification network to extract more representative and specific features based on segmented results instead of the whole dermoscopy images, further alleviating the insufficiency of training data. The proposed framework is extensively evaluated on ISBI 2016 Skin Lesion Analysis towards Melanoma Detection Challenge dataset. Experimental results demonstrate the significant performance gains of the proposed framework, ranking the first in classification and the second in segmentation among 25 teams and 28 teams, respectively. This study corroborates that very deep CNNs with effective training mechanisms can be employed to solve complicated medical image analysis tasks, even with limited training data.

Paper 11:

Melanoma is the deadliest form of skin cancer. While curable with early detection, only highly trained specialists are capable of accurately recognizing the disease. As expertise is in limited supply, automated systems capable of identifying disease could save lives, reduce unnecessary biopsies, and reduce costs. Toward this goal, proposed [11] a system that combines recent developments in deep learning with established machine learning approaches, creating ensembles of methods that are capable of segmenting skin lesions, as well as analyzing the detected area and surrounding tissue for melanoma detection. The system is evaluated using the largest publicly available benchmark dataset of dermoscopic images, containing 900 training and 379 testing images. New state-of-the-art performance levels are demonstrated, leading to an improvement in the area under receiver operating characteristic curve of 7.5% (0.843 versus 0.783), in average precision of 4% (0.649 versus 0.624), and in specificity measured at the clinically relevant 95% sensitivity operating point 2.9 times higher than the previous state of the art (36.8% specificity compared to 12.5%). Compared to the average of eight expert dermatologists on a subset of 100 test images, the proposed system produces a higher accuracy (76% versus 70.5%), and specificity (62% versus 59%) evaluated at an equivalent sensitivity (82%).

Chapter 3

Methodology

3.1Introduction

Dermatological Images and Datasets

High-quality images of skin diseases are important for both dermatologists and automated diagnostic systems. On the one hand, dermatologists rely on high-resolution (HR) images to make diagnoses when direct observation is impossible . This is especially common in telemedicine, medical consultations, and regular clinics . On the other hand, training reliable algorithms has always necessitated the use of high-quality data. In particular, deep learning algorithms always need a vast volume of labeled data for a better accuracy . As a result, high-quality dermatological images are critical for both clinical diagnosis and the design of new algorithms. In this section, we go over three different types of images commonly used in skin cancer diagnosis, as well as some public datasets.

2.1 Dermatological Images

The three main types of image modalities used to diagnose skin diseases are clinical images, dermoscopy images, and histopathological images (see Figure 1). Clinical images are frequently captured by mobile devices for remote diagnosis or as medical records. Dermoscopy images and histopathological images are commonly utilized in clinical diagnosis to assess the severity of the illness. In the next part, we introduce them separately.

FIGURE 1 Examples of three types of dermatological images of BCC to show their differences and relationships: **(A)** Clinical image. **(B)** Dermoscopy image. **(C)** Histopathological image.[1]

3.1.1 Clinical Images

Clinical images are obtained by photographing the skin disease site directly with a camera. They can be used as a medical record for patients and provide different insights for dermoscopy images . The biggest issue of utilizing clinical images for skin cancer classification is that they include limited morphological information while also introducing considerable inaccuracies into the diagnostic results, owing to the effect of diverse imaging settings (such as lighting, angle, and so on.

3.1.2 Dermoscopy Images

Dermoscopy images are captured with dermoscopy, a type of optical observation tool used to assess the fine details of skin diseases (41). Clinicians frequently utilize dermoscopy to diagnose benign nevi and malignant melanoma (42). It serves as a bridge between clinical and pathological aspects, and thus dermoscopy is often referred to as a dermatologist's stethoscope. Dermoscopy images provide a clear visualization of the skin's surface and are used to analyze the color and microstructure of the epidermis (43). For some skin diseases, there are already numerous diagnostic guidelines based on dermoscopy images (44), for example, the ABCD Rule Law (15), the CASH Rule Law (45), and the Menzies Method (16). When using dermoscopy images for skin cancer diagnosis, the range of structures that can be observed is limited, and its diagnostic accuracy is occasionally affected by the experience of dermatologists (46).

3.1.3 Histopathological Images

Histopathological images were obtained using microscopes to scan tissue slides and then digitalize as images (28). They are utilized to show the vertical structure and complete internal characteristics of the diseased tissue. In the clinic, pathological examinations serve as the "gold standard" for diagnosing almost all types of cancers, as they are often used to distinguish between types of cancers and guide appropriate treatment plans based on pathological changes. However, different histopathological images of skin cancer exhibit different morphologies, scales, textures, and color distributions, which makes it difficult to find a common pattern for diagnosis

GENERAL STEPS/MODEL FOR EARLY DETECTION OF SKIN CANCER:

This section describes about the method which is utilized to detect the melanoma and provides a detailed modeling. In recent time, the number of melanoma patients has been drastically enhanced across all over the globe due to extensive global warming. The melanoma skin cancer especially found in United States and Australia and a major part of these countries are affected with melanoma skin cancer. According to a survey of World health Organization (WHO), around 13 million people become affected each and every year by deadly melanoma skin cancer [17]. Therefore, due to extensive mortality rate across the globe and higher medical diagnosis cost, the detection of melanoma skin cancer at earliest stages is becomes a mandatory requirement. Therefore, to efficiently detect melanoma in early stages and to diagnose the melanoma skin cancer, various steps is shown below.

Step 1: Image Acquisition for Screening Skin Lesions:

Unaided visual inspection of the skin is often suboptimal for diagnosing melanoma. Numerous imaging modalities are under investigation to determine their usefulness in imaging and ascertaining a correct in vivo diagnosis of melanoma. These include total cutaneous photography, dermoscopy, confocal scanning laser microscopy (CSLM), ultrasound, magnetic resonance imaging (MRI), optical coherence tomography (OCT), and multispectral imaging.

Step 2: Preprocessing:

The main processing step towards a complete analysis of pigmented skin lesion is to differentiate the lesion from the healthy skin. Detection of the lesion is a difficult problem in dermatoscopic images as the transition between the lesion and the surrounding skin is smooth and even for trained dermatologist; it is a challenge to distinguish accurately. It has been observed that dermoscopy images often contain artifacts such as uneven illumination, dermoscopic gel, black frames, ink markings, rulers, air bubbles, and intrinsic cutaneous features that can affect border detection such as blood vessels, hairs, and skin lines and texture.

Step 3: Segmentation:

Segmentation refers to the partitioning of an image into disjoint regions that are homogeneous with respect to a chosen property such as luminance, color, and texture. The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze.

However, for the development of automated diagnostic system for skin lesion detection, it is very important to develop automatic segmentation algorithms. As segmentation is a crucial early step in the analysis of lesion images, it has become one of the important areas of research and many algorithms and segmentation techniques are available in the literature.

Step 4: Feature Extraction:

The purpose of feature extraction is to reduce the original data set by measuring certain properties, or features, that differentiate one input pattern from another. The feature extraction is performed by measurements on the pixels that represent a segmented object allowing various features to be computed. Unfortunately, the feature extraction step is often subject to error. In most of the publications dealing with this topic, many features are extracted to feed a sophisticated classifier, but there is very little discussion about the real meaning of those features and about objective ways to measure them.

Step 5: Classification:

Classification phase of the diagnostic system is the one in charge of making the inferences about the extracted information in the previous phases in order to be able to produce a diagnostic about the input image. There are two different approaches for the classification of dermoscopic images: the first considers only a dichotomous distinction between the two classes (melanoma and benign) and assigns class labels 0 or 1 to data item. The second attempts to model, this yields not only a class label for a data item, but also a probability of class membership. The most prominent representatives of the first approach are support vector machines. Logistic regression, artificial neural networks, -nearest neighbours, and decision trees are all members of the second approach, although they vary considerably in building an approximation to from data**.**

PROBLEM IDENTIFIED:

Even though we have been educated on the dangers of skin cancer and its prevention, Melanoma/skin cancer continues to increase at an alarming rate. It is the fact that skin melanoma has become one of the fastest increasing cancers in Western countries. This is due to in the case of the melanoma epidemic, a systematic analysis according to the standard problem solving procedures seems lacking in the medical scientific literature. Instead it is just assumed that changed sun-tanning habits were the true cause and that no deeper analysis of reasons behind this national catastrophe was necessary.

Skin melanoma is a cancer that normally takes many years to develop. Initial skin damages, e.g., due to DNA damage caused by UV-radiation from the sun, may develop decades later into melanoma in the skin and then also spread to other parts of the body. Under normal conditions the body can cope with such damages quite well, and damages older than, say, 30 years should have been repaired or disposed by the immune system. The melanoma problem is characterized by a suddenly increasing incidence from about 1955 in the Nordic countries. If increasing rates were caused by increasing skin damages from UV radiation, this exposure must have started almost stepwise some decades earlier affecting all ages. At the beginning of the 20th century, melanoma was mainly found on sun-exposed areas of the body such as on the head or sometimes even on the feet. But after 1955 melanoma increased fastest on normally non-exposed parts of the body. In Sweden the incidence of melanoma has been higher at lower latitudes and less in the northern parts of the country. The highest incidence has been noticed in a municipality in the middle of the country of Finland, Norway and Sweden. In all three countries melanoma seems higher in coastal areas and especially in the southern parts. These areas are also more densely populated and urban compared to the more rural areas.

However, Melanoma can be completely cured if it is detected at the early stages. Therefore, automated skin segmentation becomes highly essential for the detection of Melanoma. However, various state-of-at-techniques consist of different problems, which degrades their performance and creates delay and faulty results while diagnosis of melanoma. The problems which are occurred in the existing state-of-art-techniques are the pixel-wise prediction problem which can be formulated as a linear prediction, which can be efficiently implemented in a coarse-to-fine manner by upsampling coarse predictions (bilinear interpolation through “deconvolution”). The second problem is that the independent prediction of the class labels for the image pixels cannot incorporate the correlation between the image regions, such as the spatial smoothness of neighboring pixels, which is crucial for distinguishing lesions from the background skin. Another problem which is often occurs is the optimal minimization problem and Centre-surround contrast problem when a salient region is located at the image boundary and when a foreground region is globally compared to the rest of the image. Image classification, object detection and segmentation problems, insufficient training, overlapping problems, object detection related problems and vascular segmentation problems are the main problems which often while segmenting the melanoma

Existing works have shown evidence that the problem of insufficient training data can be alleviated by fine-tuning, where the lower layers of the fine-tuned network are more general filters (trained on general images) while those in the higher layers are more specific to the target problem

3.2 Hardware and Software Requirements

Hardware Requirements

- Processor: Dual core above
- RAM:2 GB RAM [minimum]
- Monitor: 15 INCH LED
- Keyboard: Standard 120 keys
- Mouse: ANY
- Hard disk:120GB [minimum]

Software Requirements

- Operating system: Windows
- Technology:Python
- Tools:
- Front end: Python
- Backend: Python

TECHNOLOGY USED

Software selection is the critical aspect of system development the search starts with the software. There are a number of criteria for software selection.

PYTHON:

Created by Guido van Rossum and first released in 1991. Python name is derived from the British comedy group Monty Python, whom Python creator Guido van Rossum enjoyed while developing the language. Python is an interpreter, object oriented, high-level programming language with dynamic semantics. Its high level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed. Often, programmers fall in love with Python because of the increased productivity it provides.

Since there is no compilation step, the edit-test-debug cycle is incredibly fast. The latest version of python is 3.7.2. Python is a mutli-paradigm programming language. Object oriented programming and structured programming are fully supported. It uses dynamic typing, and a combination of reference counting and a cycle-detecting garbage collector for memory management. It also features late binding. It is highly extensible. Its formatting is visually uncluttered and it often uses English keywords. Unlike other languages, it does not use curly brackets to delimit blocks and semicolons after statements are optional. It uses whitespace indentation to delimit blocks. Since 2003, Python has consistently ranked in the top ten most popular programming languages in the TIOBE Programming Community Index where, as of December 2018, it is the third most popular language (behind Java, and C). It was selected Programming Language of the Year in 2007, 2010, and 2018. Large organizations that uses python include Wikipedia, Google, Yahoo,CERN, NASA, Facebook, Amazon, Instagram, Spotify. And some smaller entities like ILM and ITA. The social news networking sites Reddit is written entirely in python.

ANACONDA:

It is a free and open source distribution of the Python and R programming languages for scientific computing (data science, machine learning applications, large-scale data processing, predictive analytics, etc.), that aims to simplify package management and deployment. It was formerly known as Continuum Analytics. Package versions are managed by the package management system conda.

The Anaconda distribution is used by over 13 million users and includes more than 1400 popular data-science packages suitable for Windows, Linux, and MacOS. Anaconda Navigator.

Anaconda Navigator is a desktop graphical user interface(GUI) included in Anaconda distribution that allows users to launch applications and manage conda packages, environments and channels without using command-line commands.

Conda is an open source, cross-platform, language-agnostic package manager and environment management system that installs, runs, and updates packages and their dependencies. It was created for Python programs, but it can package and distribute software for any language (e.g., R), including multi-language projects. The Conda package and environment manager is included in all versions of Anaconda, Miniconda, and Anaconda Repository.

Anaconda Cloud is a package management service by Anaconda where you can find, access, store and share public and private notebooks, environments, and conda and PyPI packages.

Cloud hosts useful Python packages, notebooks and environments for a wide variety of applications. One do not need to log in or to have a Cloud account, to search for public packages, download and install them.One can build new packages using the Anaconda Client command line interface (CLI), then manually or automatically upload the packages to Cloud.

Anaconda can help with :

(i) Installing Python on multiple platforms
(ii) (ii) Separating out different environments
(iii) (iii) Dealing with not having correct privileges and
(iv) (iv) Getting up and running with specific packages and libraries.

The following applications are available by default in Navigator:

(i) JupyterLab :
JupyterLab is the next-generation web-based user interface for Project Jupyter. It enables you to work with documents and activities such as Jupyter notebooks, text editors, terminals, and custom components in a flexible, integrated, and extensible manner.

(ii) Jupyter Notebook:
The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text. Uses include:

data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more.

(iii) QtConsole:

QtConsole, which melds the feel of a lightweight terminal and the functionality of an advanced GUI editor, is a rich interface to Jupyter kernels that can be used standalone or built in to an IDE like Spyder (with which QtConsole is a joint collaboration).

(iv) Spyder:

the Scientific Python Development Environment, is a free integrated development environment (IDE) It includes editing, interactive testing, debugging and introspection features.

(v) Glueviz:

Glue is a multi-disciplinary tool. Designed from the ground up to be applicable to a wide variety of data, Glue is being used on astronomy data of star forming-clouds, medical data including brain scans, and many other kinds of data.

(vi) Orange:

Open source machine learning and data visualization for novice and expert is an Interactive data analysis workflows with a large toolbox.

(vii) Rstudio: RStudio provides popular open source and enterprise-ready professional software for the R statistical computing environment.

(viii) Visual Studio Code: Visual Studio Code is a source-code editor developed by Microsoft for Windows, Linux and macOS. It includes support for debugging, embedded Git control, syntax highlighting, intelligent code completion, snippets, and code refactoring.

NUMPY

NumPy is the fundamental package for scientific computing with Python. It contains among other things: a powerful N-dimensional array object, sophisticated (broadcasting) functions, tools for integrating C/C++ and Fortran code and useful linear algebra, Fourier transform, and random number capabilities. Besides its obvious scientific uses, NumPy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined. This allows NumPy to seamlessly and speedily integrate with a wide variety of databases.

PANDAS:

Pandas is an open source, BSD-licensed library providing high-performance, easy-to-use data structures and data analysis tools for the Python programming language. Pandas is the most popular python library that is used for data analysis. It provides highly optimized performance with back- end source code is purely written in C or Python.

SCIKIT-LEARN:

Scikit-learn is a package that enables Machine Learning in Python. It provides Simple and efficient tools for data mining and data analysis. Its major advantage is that it is accessible to everybody, and reusable in various contexts. Scikit-learn is built on NumPy, SciPy, and matplotlib. It is open source and commercially usable. It features various classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting, k-means and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.

MATPLOTLIB:

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK+.Pyplot is a Matplotlib module which provides a MATLAB-like interface. SciPy makes use of Matplotlib

3.3 Modules and Descriptions

The system comprises of 3 major modules and their sub modules as follows:

1. Admin:

In this module, the admin can view users, and update and delete the details of each user.

2. Authentication

Here, we can upload the images from user and the prediction of skin cancer can be viewed.

3. Image Clustering: The data sets are collected from ISIC(International Skin Image Collaboration) which provides the images for checking accuracy. The datasets contain approximately 23000 images of which we have collected 1000-1500 image and trained and tested over the images.

4. Image processing : This module involves the preprocessing of the images where hair removal, glare removal and shading removal are done. Removal of these parameters helps us to identify the texture,color,size and shape like parameter in an efficient way.

5. User Data Processing: The users have to register and upload the captured image for accuracy check.The image is then compared with the standard data set and the results are provide.

6. Alert and notification: The stage of skin cancer and the results are provided to the registered user as a message or an email via the registered number or registered mail Id.

Software Environment

Python:

- Python is a high-level, interpreted, interactive and object-oriented scripting language.
- Python is designed to be highly readable.
- It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.
- Python is Interpreted − Python is processed at runtime by the interpreter.

- You do not need to compile your program before executing it. This is similar to PERL and PHP.
- Python is Interactive − You can actually sit at a Python prompt and interact with the interpreter directly to write your program.
- Python is Object-Oriented − Python supports Object-Oriented style or technique of programming that encapsulates code within objects
- Python is a Beginner's Language –
- Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

History of Python

Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands. Python is derived from many other languages, including ABC, Modula-3, C, C++, Algol-68, SmallTalk, and Unix shell and other scripting languages. Python is copyrighted. Like Perl, Python source code is now available under the GNU General Public License (GPL). Python is now maintained by a core development team at the institute, although Guido van Rossum still holds a vital role in directing its progress.

Python Features

Python's features include –

- Easy-to-learn − Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.
- Easy-to-read − Python code is more clearly defined and visible to the eyes.
- Easy-to-maintain − Python's source code is fairly easy-to-maintain.
- A broad standard library − Python's bulk of the library is very portable and crossplatform compatible on UNIX, Windows, and Macintosh.
- Interactive Mode − Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
- Portable − Python can run on a wide variety of hardware platforms and has the same interface on all platforms.
- Extendable − You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.
- Databases − Python provides interfaces to all major commercial databases.

- GUI Programming − Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.
- Scalable − Python provides a better structure and support for large programs than shell scripting. Apart from the above-mentioned features, Python has a big list of good features, few are listed below –
- It supports functional and structured programming methods as well as OOP.
- It can be used as a scripting language or can be compiled to byte-code for building large applications.
- It provides very high-level dynamic data types and supports dynamic type checking.
- It supports automatic garbage collection.
- It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java. Python is available on a wide variety of platforms including Linux and Mac OS X. Let's understand how to set up our Python environment.

Getting Python.

The most up-to-date and current source code, binaries, documentation, news, etc., is available on the official website of Python https://www.python.org. Windows Installation Here are the steps to install Python on a Windows machine.

- Open a Web browser and go to https://www.python.org/downloads/.
- Follow the link for the Windows installer python-XYZ.msifile where XYZ is the version you need to install.
- To use this installer python-XYZ.msi, the Windows system must support Microsoft Installer 2.0. Save the installer file to your local machine and then run it to find out if your machine supports MSI.
- Run the downloaded file. This brings up the Python install wizard, which is easy to use. Just accept the default settings, wait until the install is finished, and you are done.

The Python language has many similarities to Perl, C, and Java. However, there are some definite differences between the languages. What is Python? Python is a popular programming language. It was created in 1991 by Guido van Rossum. It is used for:

- web development (server-side),
- software development,
- mathematics,
- System scripting. Why Python?
- Python works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc).
- Python has a simple syntax similar to the English language.
- Python has a syntax that allows developers to write programs with fewer lines than some other programming languages.
- Python runs on an interpreter system, meaning that code can be executed as soon as it is written. This means that prototyping can be very quick.

Implementation Platform and Language

Proposed System

A CNN model using images from the image data store is presented schematically to generate discriminative and relevant attribute interpretations for the cancer detection technique, as shown in Algorithm 1. To begin, a basic explanation of the used dataset is provided. Moreover, the details of the implementation of proposed model, including preprocessing techniques and the basic architecture, are presented.

Data are at the core of DL, representing what these learning techniques run on. Cancer is a unique disease, and there have already been many datasets published. We used lesion images from publicly accessible image databases of identified affected individuals. The ISIC 2018 dataset was utilized for training the proposed approach, which contained 10,015 training and 1512 test images for a total of 11,527 images [30]. ISIC 2018 provided the ground-truth data only for the training set, consisting of seven classes, melanoma, melanocytic nevus, basal cell carcinoma, squamous cell carcinoma, vascular lesions, dermatofibroma, and benign keratosis, as shown in Figure.

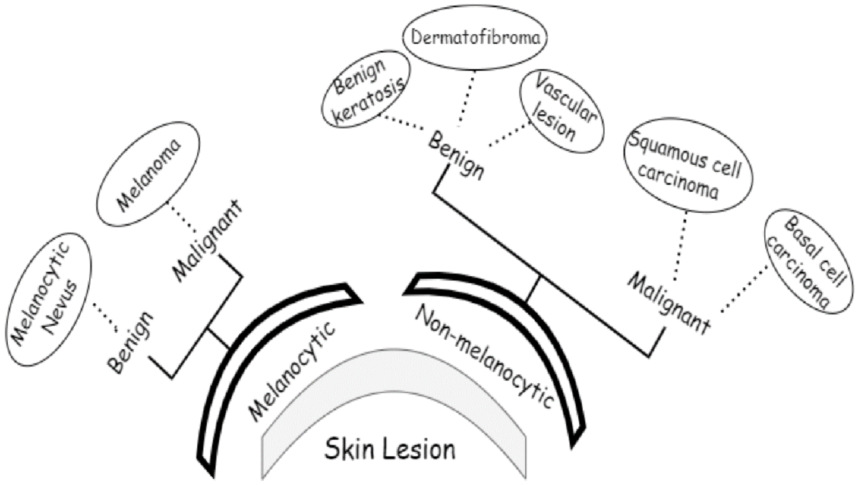


Fig: 3. Classes of ISIC2018 dataset.

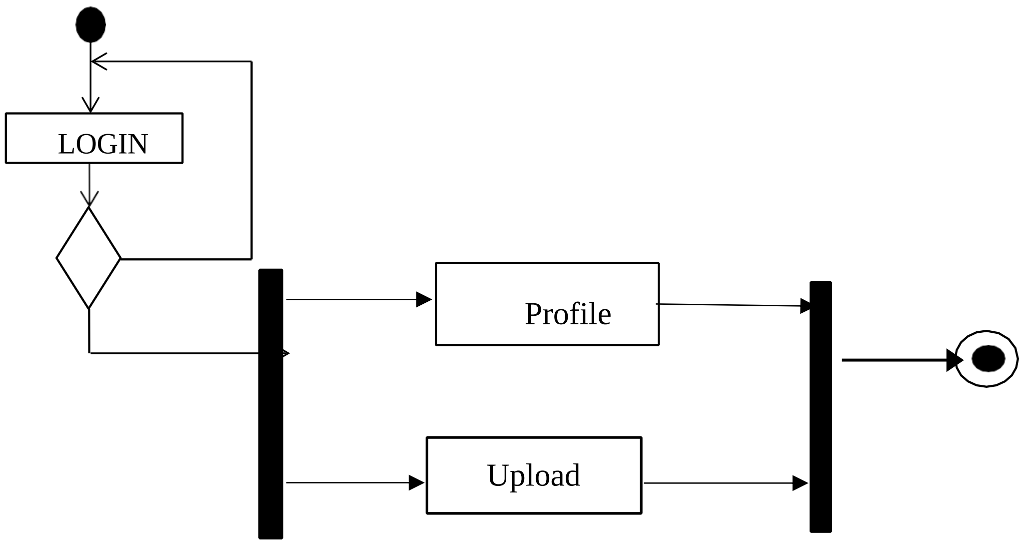
We applied the proposed CNN model to the ISIC 2018 skin lesion classification challenge test set; our data store consisted of 3533 lesion scans where 1760 of them are benign and 1773 are malignant, and we tested the proposed system using a total of 960 images consisting of 360 benign and 300 malignant cases. The lesion images were acquired from an openly accessible data repository ISIC 2018 [31]. For evaluation, the authors obtained radiological scans from many legitimate databases of cancer incidences; images from this source are used in most cancer diagnostics. The database, which is updated regularly, offers a free library of cancer cases and lesion images. The Kaggle list "Lesion Images" was used to collect lesion images; 3533 images from these sources are included in the ISIC2018 collection [41]. Figure 4 shows various lesion image examples from the ISIC2018 dataset, demonstrating the collection's diversity of patient situations. It was decided to build ISIC2018 because the library is openly available and openly available to the academic communities and the public society.

3.4 Work flow

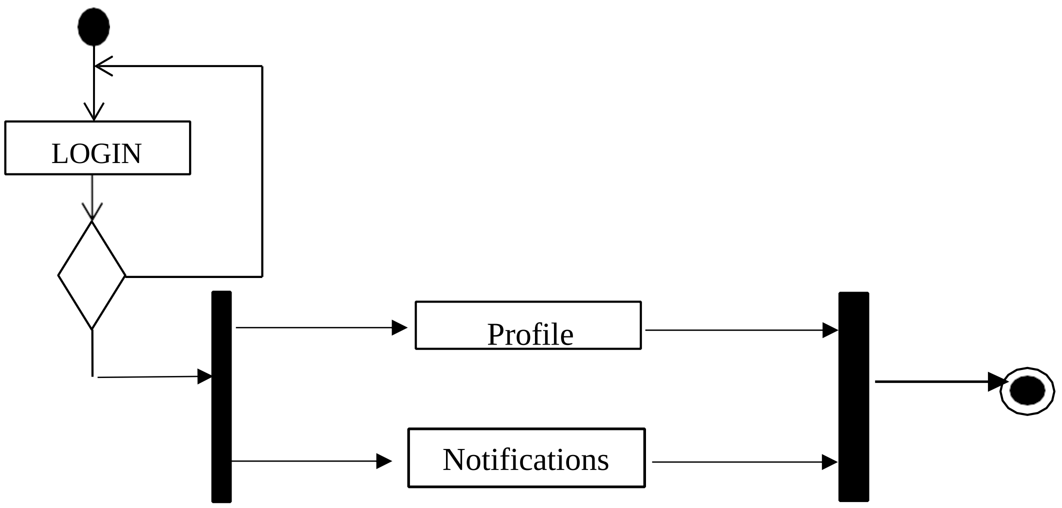
A workflow diagram is a visual layout of a process, project or job in the form of a flow chart. It's a highly effective way to impart the steps more easily in a business process, how each one will be completed, by whom and in what sequence.

ACTIVITY DIAGRAM

1) User Activity Diagram



2) Admin Activity Diagram



UML Use Case Diagrams

UML DIAGRAMS

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object oriented computer software.In its current form UML is comprised of two major components:a Meta-model and a notation.In the future,some form of method or process may also be added to or associated with, UML. The Unified Modeling Language is a standard language for specifying,Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems. The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

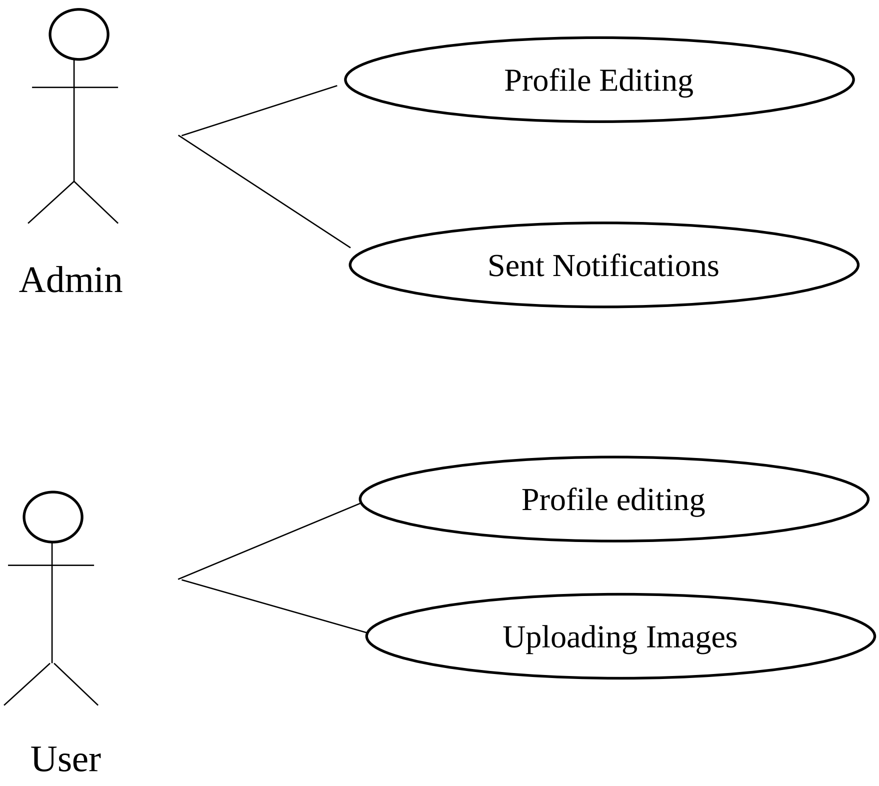
GOALS:

The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modeling language.
5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

Use Case Diagrams

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purposeof a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.


Profile Editing
Sent Notifications
Admin
Profile editing
Uploading Images
User

Chapter 4

Agile Methodology

Introduction

After the initial studies, it is found that the agile model of software development is suitable and is the best method for the development of this system. Agile methodology mainly focused on client satisfaction through continuous delivery. Also, it sets a minimum number of rerequirements and turns them into a deliverable product. As this project has many individual requirements which can be delivered in parts and the user can gradually improve their work efficiency. Agile methodology has a family of methods and scrum is selected for the development of this project. Scrum is a process framework that has been used to manage complex product development. It is not a process or technique for building products rather it is a framework within which various processes can be employed. Also, it is a suitable method to support the development process. It focuses on lean software development and has in building better software effectively and efficiently. Agile is one of the most widely used and recognized software development frameworks. The methodology those experts agreed upon was described as 'lightweight' and fast. Agile is also about being adaptive and continuous improvement, as much asit is about constant feedback and speed of delivery. Agile is a software development approach where a self-sufficient and cross-functional team works on making continuous deliveries through iterations and evolves throughout the process by gathering feedback from the end users. The major rules of scrum methodology are: 1. The product owner (PO): Who represents the stakeholder and the business. 2. The scrum master: Ensures the process followed, removes obstructions, and protects the development system 3. Development team: Cross-functional, self-organizing team who does the actual analysis, design implementation, and testing process. They work together in iterative time-boxed durations called sprints. The first step is the creation of the product backlog by the PO.It's a to-do list of stuff to be done by the scrum team. Then the scrum team selects the top priority items and tries to finish them within the time box called a sprint. An easier way to remember all of this is to memorize the 3-3-5 framework. It means that a scrum project has 3 roles, 3 artifacts, and 5 events These are :

1. Roles: Product Owner, Scrum Master, and development team.
2. Artifacts: Product Backlog, Sprint Backlog, and Product Increment.
3. Events: Sprint, Sprint planning, Daily Scrum, Sprint review, and Sprint retrospective

The framework begins with a simple premise starting with what can be seen or known. After that, the progress is tracked and tweaked as necessary. The three pillars of scrum are transparency, inspection, and adaptation. In scrum, everyone has a role.

4.2 User Story

A user story is a tool used in agile software development to capture a description of software feature from an end-user perspective. The user story describes the type of user, what they want and why. A user story helps to create a simplified description of a requirement.

User Story ID	As a <Type of user>	I want to perform <some task>	So that I can <achieve some goal>
1	USER	Register to the system	Access the system
2	USER	Login	Access the account
3	USER	Uploading the image	Get Result
4	ADMIN	Send a Message	Notify the User

Table 4.1 user story

4.3 Product Backlog

A product backlog is a list of the new features, changes to existing features, bug fixes, infrastructure changes or other activities that a team may deliver in order to achieve a specific outcome. The product backlog is the single authoritative source for things that a team works on. That means that nothing gets done that isn’t on the product backlog. Conversely, the presence of a product backlog item on a product backlog does not guarantee that it will be delivered. It represents an option the team has for delivering a specific outcome rather than a commitment.

It should be cheap and fast to add a product back log item to the product backlog, and it should be equally as easy to remove a product backlog item that does not result in direct progress to achieving the desired outcome or enable progress toward the outcome. The Scrum Product Backlog is simply a list of all things that needs to be done within the project. It replaces the traditional requirements specification artifacts. These items can have a technical nature or can be user-centric e.g., in the form of user stories. The product backlog of the system is given in Table 4.2

PRODUCTBACKLOG			
ID	Name	Priority	Estimate[Hrs]
1	Registration	1	10
2	Login	2	10
3	Upload the Image File	3	10
4	Predictive Analysis of the Product	4	15

Table4.2: Product Backlog

4.4 Project Plan

A project plan that has a series of tasks laid out for the entire project, listing task durations, responsibility assignments, and dependencies. Plans are developed in this manner based on the assumption that the Project Manager, hopefully along with the team, can predict up front everything that will need to happen in the project, how long it will take, and who will be able to do it. Project plan is given in Table 4.3

User story ID	Task Name	Start Date	End Date	Days	Status
Sprint 1		18-8-2022	7-9-2022	**19**	**Completed**
1	Registration	18-8-2022	22-8-2022	4	Completed
2	Coding	23-8-2022	30-8-2022	8	Completed
3	Testing	31-8-2022	7-9-2022	7	Completed

Sprint 2		13-9-2022	8-10-2022	**24**	**Completed**
7	coding	13-9-2022	26-9-2022	13	Completed
8	Testing	27-9-2022	8-10-2022	11	Completed
Sprint 3		13-10-2022	31-10-2022	**17**	**Completed**
9	Database Connectivity	13-10-2022	23-10-2022	10	Completed
10	Uploading Image	24-10-2022	31-10-2022	7	Completed
Sprint 4		5-11-2022	22-11-2022	**16**	**Completed**
12	Deployment	5-11-2022	13-11-2022	8	Completed
13	Testing and Validation	14-11-2022	22-11-2022	8	Completed

Table4.3: Project plan

The Project has four sprints:

1. **Sprint 1**

Three tasks are planned in this sprint. First one is Problem definition, next is designing and initial coding

2. **Sprint 2**

Three tasks are planned in this sprint. First one is design and development of forms and next one is testing.

3. **Sprint 3**

Two tasks are planned in this sprint. These are upload images and development of web user interface using Django and Convolutional Neural Network. The final task is the prediction.

4. **Sprint 4**

In this sprint two tasks are planned to complete, one is Deployment of web app and Booking and second is testing and result discussion.

4.5 Sprint Backlog (plan)

The sprint backlog is a list of tasks identified by the Scrum team to be completed during the Scrum sprint. During the sprint planning meeting, the team selects some number of product back log items, usually in the form of user stories, and identifies the tasks necessary to complete each user story. Most teams also estimate how many hours each task will take someone on the team to complete.

1. **Sprint 1**

Three tasks are planned in this sprint. First one is Problem definition , next is designing and initial coding

2. **Sprint 2**

Three tasks are planned in this sprint. First one is design and development of forms and next one is testing.

3. **Sprint 3**

Two tasks are planned in this sprint. These are upload images and development of web user interface using Django and Convolutional Neural Network. The final task is the prediction.

4 . **Sprint 4**

In this sprint two tasks are planned to complete, one is Deployment of web app and Booking and second is testing and result discussion.

Table 4.4: Sprint Backlog(Plan)-Sprint 1

Backlog item	Completion time	Original estimate in hours	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Day 8	Day 9	Day 10	Day 11	Day 12	Day 13	Day 14
User Story #1 Hours			hours	hours	hours	hours	hours	hours	hours	hours	hours	hours	hours	hours	hours	hours
Registration	22-8-2022	6	1	1	1	1	1	1	0	0	0	0	0	0	0	0
Coding	30-8-2022	4	1	0	0	0	0	1	1	1	0	0	0	0	0	0
Testing	7-9-2022	4	0	0	0	0	0	0	1	1	1	1	0	0	0	0
Total		14	2	1	1	1	1	2	2	2	1	1	0	0	0	0

Bac klog ite m	Co m ple tion time	Or igi na l esti mat e in ho rs	D ay 1	D ay 2	D ay 3	D ay 4	D ay 5	D ay 6	D ay 7	D ay 8	D ay 9	D ay 10	D ay 11	D ay 12	D ay 13	D ay 14
User Stor y #1 Hou rs			ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs
Codi ng	26-9-2022	7	1	1	1	1	1	1	1	0	0	0	0	0	0	0
Testi ng	8-10-2022	6	0	0	0	0	0	0	0	0	1	1	1	1	1	1
Total		13	1	1	1	1	1	1	1	0	1	1	1	1	1	1

Table 4.6: Sprint Backlog (Plan)-Sprint 2

Bac klog ite m	Co m ple tion time	Or igi na l esti mat e in ho rs	D ay 1	D ay 2	D ay 3	D ay 4	D ay 5	D ay 6	D ay 7	D ay 8	D ay 9	D ay 10	D ay 11	D ay 12	D ay 13	D ay 14
User Stor y #1 Hou rs			ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs
db conn ectiv ity	23-10-2022	10	1	1	1	1	1	1	0	1	1	1	1	0	0	0
Upload Image	31-10-2022	7	1	1	1	0	0	1	0	1	0	1	0	1	0	0
Tota l		17	2	2	2	1	1	2	0	2	1	2	1	1	0	0

Table 4.7: Sprint Backlog (Plan)-Sprint 3

Backlog Item	Completion time	Original estimate in hours	Day 1	Day 2	Day 3	Day 4	Day 5
User Story #1 Hours							
Development	13-11-2022	6	2	0	2	2	0
Testing and Validation	22-11-2022	4	2	0	0	1	1
Total		10	4	0	2	3	1

Table 4.8: Sprint Backlog (Plan)-Sprint 4

4.6 Sprint Backlog (Actual)

Actual sprint backlog is what adequate sprint planning is actually done by project team there may or may not be difference in planned sprint backlog. The detailed sprint backlog(Actual)is given below.

Backlog item	Completion time	Original estimate in hours	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Day 8	Day 9	Day 10	Day 11	Day 12	Day 13	Day 14
User Story #1 Hours			hours	hours	hours	hours	hours	hours	hours	hours	hours	hours	hours	hours	hours	hours
Registration	22-8-2022	6	1	1	1	1	1	1	0	0	0	0	0	0	0	0
Coding	30-8-2022	4	1	0	0	0	0	1	1	1	0	0	0	0	0	0
Testing	7-9-2022	4	0	0	0	0	0	0	1	1	1	1	0	0	0	0
Total		14	2	1	1	1	1	2	2	2	1	1	0	0	0	0

Table 4.9: Sprint Backlog(Actual)-Sprint1

Backlog ite m	Com ple tion time	Origi na l esti mat e in ho rs	D ay 1	D ay 2	D ay 3	D ay 4	D ay 5	D ay 6	D ay 7	D ay 8	D ay 9	D ay 10	D ay 11	D ay 12	D ay 13	D ay 14
User Story #1 Hou rs			ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs
Codi ng	26-9-2022	7	1	1	1	1	1	1	1	0	0	0	0	0	0	0
Testi ng	8-10-2022	6	0	0	0	0	0	0	0	0	1	1	1	1	1	1
Total		13	1	1	1	1	1	1	1	0	1	1	1	1	1	1

Table 4.11: Sprint Backlog (Actual)-Sprint 2

Bac klog ite m	Co m ple tion time	Or igi na l esti mat e in ho rs	D ay 1	D ay 2	D ay 3	D ay 4	D ay 5	D ay 6	D ay 7	D ay 8	D ay 9	D ay 10	D ay 11	D ay 12	D ay 13	D ay 14
User Stor y #1 Hou rs			ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs	ho urs
db conn ectiv ity	23-10-2022	10	1	1	1	1	1	1	0	1	1	1	1	0	0	0
Upload Image	31-10-2022	7	1	1	1	0	0	1	0	1	0	1	0	1	0	0
Tota l		17	2	2	2	1	1	2	0	2	1	2	1	1	0	0

Table 4.12: Sprint Backlog (Actual)-Sprint 3

Backlog Item	Completion time	Original estimate in hours	Day 1	Day 2	Day 3	Day 4	Day 5
User Story #1 Hours							
Development	13-11-2022	6	2	0	2	2	0
Testing and Validation	22-11-2022	4	2	0	0	1	1
Total		10	4	0	2	3	1

Table 4.13: Sprint Backlog (Actual)-Sprint 4

4.7 Product Backlog Review

REVIEW FORM

Sprint 1

Version: 1.0 Date:7-9-2022

User Story ID	Comments from Scrum master if any	Comments from Product Owner if any
1	Developer should have a easy Login process	User friendly registration
2	effective login	if there is forgot password or username handled.

Table 4.9: Product Backlog Review (Sprint1)

Sprint 2

Version:1.0 Date: 8-10-2022

User Story ID	Comments from Scrum master if any	Comments from Product Owner if any
3	Should check the data anatomize correctly	Inserted
4	Design and development of registration forms	Should customize different models

Table 4.10: Product Backlog Review (Sprint2)

Sprint 3

Version: 1.0 Date: 31-10-2022

User Story ID	Comments from Scrum master if any	Comments from Product Owner if any
5	should check database connectivity	Check connection.
6	Melanoma prediction	Check prediction

Table 4.12: Product Backlog Review (Sprint3)

Sprint 4

Version: 1.0 Date: 22-11-2022

User Story ID	Comments from Scrum master if any	Comments from Product Owner if any
7	Deployment	Visualize final output.
8	Generate predicted result.	Satisfied.

Table 4.13: Product Backlog Review (Sprint4)

4.8 Sprint Review:

At the end of each sprint a Sprint Review meeting is held. During this meeting the Scrum Team shows which Scrum Product Backlog items they completed (according to the Definition of Done) during the sprint. This might take place in the form of a demo of the new features. Backlog items that are not completed shall not be demonstrated. Otherwise this might suggest that these items are finished as well. Instead incomplete items/remaining activities shall be taken back into the Scrum Product Backlog, re-estimated and completed in one of the following sprints. The Sprint Review meeting should be kept very informal. No Power Point slides should be used and time for preparation and performing the meeting should be limited. During the meeting the Scrum Product Owner inspects the implemented backlog entries and accepts the solution or adds new stories to the Scrum Product Backlog to adapt the functionality. Participants in the sprint review typically include the Scrum Product Owner, the Scrum Team and the Scrum Master. Additionally management, customers, and developers from other projects might participate as well.

REVIEW FORM

SPRINT 1

Version: 1.0 **Date: 7-9-2022**

User story ID	Comments from Scrum master if any	Comments from Product Owner if any
1	Developer should have a easy Login process	Satisfied
2	effective login	Successful

Table 4.14: Sprint Review (Sprint1)

SPRINT 2

Version: 1. 0 **Date: 8-10-2022**

User story ID	Comments from Scrum master if any	Comments from Product Owner if any
3	Should check the data inserted correctly	Correctly Successful.
4	Design and development of registration forms.	Satisfied.

Table 4.16: Sprint Review (Sprint2)

SPRINT 3

Version: 1. **Date: 31-10-2022**

User story ID	Comments from Scrum master if any	Comments from Product Owner if any
5	Should check database connectivity.	Connection successful.
6	Upload images	Successful.

Table 4.17: Sprint Review (Sprint3)

SPRINT 4

Version: 1.0 **Date: 22-11-2022**

User story ID	Comments from Scrum master if any	Comments from Product Owner if any
9	Deployment completed	Satisfied
10	Output generated	Satisfied with result

Table 4.18: Sprint Review (Sprint4)

4.9 Testing and Validation

SPRINT 1

Version: 1.0 **Date: 7-9-2022**

Test #	Date	Action	Expected Result	Actual Result	Pass? <Yes/no>
1		Registration	Registration successful	Successful	Yes
2		Login	Login to system	Login to successful	Yes

Table 4.19: Testing and Validation (Sprint1)

SPRINT 2

Version: 1.0 **Date: 8 -10-2022**

Test #	Date	Action	Expected Result	Actual Result	Pass? <Yes/no>
1		Development of models	Can choose the best models	Done	Yes

Table 4.21: Testing and Validation (Sprint2)

SPRINT 3

Version: 1.0 **Date: 31-10-2022**

Test #	Date	Action	Expected Result	Actual Result	Pass? <Yes/no>
1		Development of web application	UI will be formed	Done	Yes

Table 4.22: Testing and Validation (Sprint4)

SPRINT 4

Version: 1.0 **Date: 22-11-2022**

Test #	Date	Action	Expected Result	Actual Result	Pass? <Yes/no>
1		Deployment	Melanoma predicted Successfully	Done	Yes

Table 4.23: Testing and Validation (Sprint5)

4.10 Git

Git is a free and open-source distributed version control system designed to handle everything from small to very large projects with speed and efficiency. To show the continuous development of the project the Git-lab histories are shown in Appendix from figure A.1

The Git is used as the version control system for this project. Version control is a system that records changes to a file or set of files over time so that a specific version can be recalled later. Version control systems are a category of software tools that help a software team for managing changes to source code over time. Version control software keeps track of every modification to the code in a special kind of database. If a mistake is made, developers can turn back the clock and compare earlier versions of the code to help fix the mistake while minimizing disruption to all team members.

Fig A.1. Git graph

Chapter 5

Result And Discussions

In this project, it is seen that using deep learning there is no need of complex preprocessing techniques except for normalizing the pixel value, resizing and cropping. It bypasses all the preprocessing, segmentation and handcrafted feature ex-traction. It is observed that applying deep neural networks gives better result than conventional methods. For medical imaging,the most important challenge is that of acquiring labeled images. The ISBI challenge has provided with large number of datasets (ISIC Archive), which provides a common platform for researchers and academia to evaluate their work. The future works might include: using a much larger dataset to reduce the risk of overfitting and performing additional regularization tweaks and fine-tuning of hyper-parameters. In many papers, it is shown that taking both global (deep)as well as local features into account during feature extraction provides better results. However, it is still an open issue to increase the accuracy rate. The goal should be to target highest sensitivity while optimizing to increase specificity, thereby increasing the overall accuracy for practical application.

System Design and Implementation and Testing

INTRODUCTION

Design concept provides the basic criteria for design quality. Design is the meaningful representation for something to build. Design focus on the three major areas of concern: Data, architecture, interface beginning once the software requirements has been analyzed and specified, software design in the first of three activities - design code generation and test. Each activities transforms information in a manner that ultimately results in validated computer software. Design is the first step in moving from the problem domain towards the solution domain. The detailed design phase. This can be achieved by:

- Input design
- Output design

INPUT DESIGN

Input design is the process of converting user oriented input into computer based format. The goal of the design input is to make data entry as easy and free from error. In our system, we use django framework to design the forms' he input in the system is given through forms. Any surface on which information is to be entered, user interacts with the system through forms. When the data which is inputted to the system through the system. So the designer should ensure that the form is simple, accessible and easily understandable by the user. The people who communicate with the system through user interface frequently are known as end user; the design of the input screen must be according to the specification and needs of the end users.

The specification given by the end users is:

- Interaction window should be user friendly
- Easy to operate
- Provide with proper validations

The form design should be clear and enough information should be provided to guide the user to enter correct data. The design decision for handling of inputs specifies how data are accepted for computer processing. The design of input also includes specifying means by which system administrator direct the system in which the action to take. The main goal of the input design to make the data entry easier, accurate and error free. Security is provided in necessary area. Input design is designed in a simple manner without any complex name, figure, confusing fields etc. proper validation for necessary fields is also provided.In the input system, data is accepted and it can be readily used for data processing and also can be stored in a database for future use. The user provided data is been processed into the computer recognizable format from this input design. The name of the input design is as follows

- Provide data to the system
- User friendly
- Avoid errors in the data
- Making the process simple

OUTPUT DESIGN

The design of output is the most important task of any system. During output design, developers identify the type of outputs needed, and consider the necessary output controls and prototype report layouts.

Objectives of Output Design

The objectives of output design are −

- To develop output design that serves the intended purpose and eliminates the production of unwanted output.
- To develop the output design that meets the end users requirements.
- To deliver the appropriate quantity of output.
- To form the output in appropriate format and direct it to the right person.
- To make the output available on time for making good decisions

DATABASE DESIGN

The data in the system is been stored and retrieved from database deciding the database is a part of system decide whether it is a collection of interrelated data stored with minimum redundancy save quickly and efficiently the main aim is to make database access Quick and easy. Database design of the system deals about the relevant data that come into play in the system is identified. According to the relationship of data tables are designed by allowing the standard database design method. Data type of each data is defined for the optimum design of the database to have better response time to maintain data integrity to avoid redundancy to serve many uses to quick and efficiently. The general objective is to make information access easy, quick, inexpensive and flexible for the user. Database design is the most critical path of design phase. An elegant designed, welldefined database is a strong foundation for the whole system files in a relational database are called table’s column of table represent data and rows represent the records in conventional technology

SYSTEM IMPLEMENTATION AND TESTING

Implementation is a stage of theoretical design is turned into working the system. The implementation phase is used to test the development package with sample data, correcting the error identified, appearing the user of the various special facilities and features of the computerized system. It also involves the user training for minimize resistance to change and giving the new system a change to prove is worth: The successful implementation of the new system depends upon the involvement of the user.

SYSTEM IMPLEMENTATION

Implementation phase is the phase, which involves the process of converting a new system design into one operational one. It is the key stage in achieving a successful new system. Implementation is the stage of the project, where the theoretical design is turned into a working system. At this stage the main workload, the greatest upheaval and the major impact on existing practices shift to user department. If the implementation stage is not planned and controlled carefully it can cause chaos. Thus, it can be considered to be the more crucial stage in achieving a successful new stage and in giving the user confidence that the system will work and will be effective.

IMPLEMENATATION PROCEDURE

The implementation phase is less creative than system design. A system project may be dropped at any time prior to implementation although it becomes more difficult when it goes to the system phase. The final report to the implementation phase includes procedural, records layouts, reports layouts and a workable plan for implementing the candidate system. Implementation is used to the process of converting a new or revised system design into an operational one. Conversion is one aspect of implementation is unique to implementation phase.

TABLE STRUCTURE

TABLE NAME ADMIN:

S.NO	COLUMN	DATA TYPE	SIZE	CONSTRAINT	DESCRIPTION
1	Admin_id	int		Pk	Id of admin
2	Username	varchar	15	Not null	Username
3	Password	varchar	15	Not null	Password

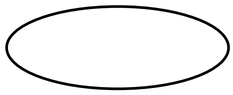
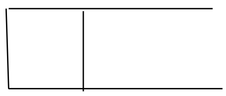
TABLE NAME USER

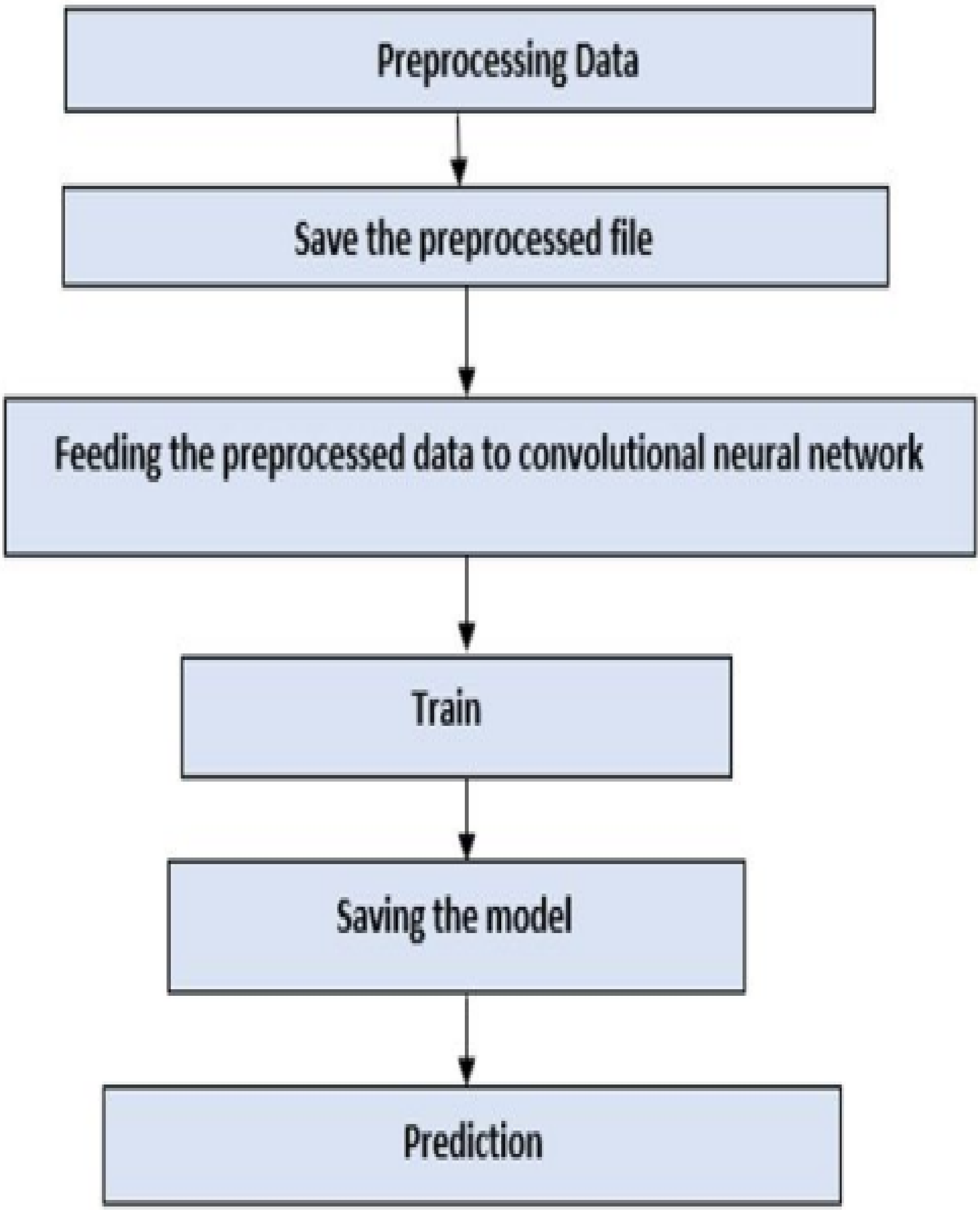
S.NO	COLUMN	DATA TYPE	SIZE	CONSTRAINT	DESCRIPTION
1	Id	Integer		pk	Id number
2	Username	Varchar	15	Not null	Username
3	Password	Varchar	15	Not null	Password
4	Name	Varchar	25	Not null	Name
5	Email	Varchar	25	Not null	Email id
6	Contact	Varchar	15	Not null	Contact Number

RESULT AND DISCUSSIONS

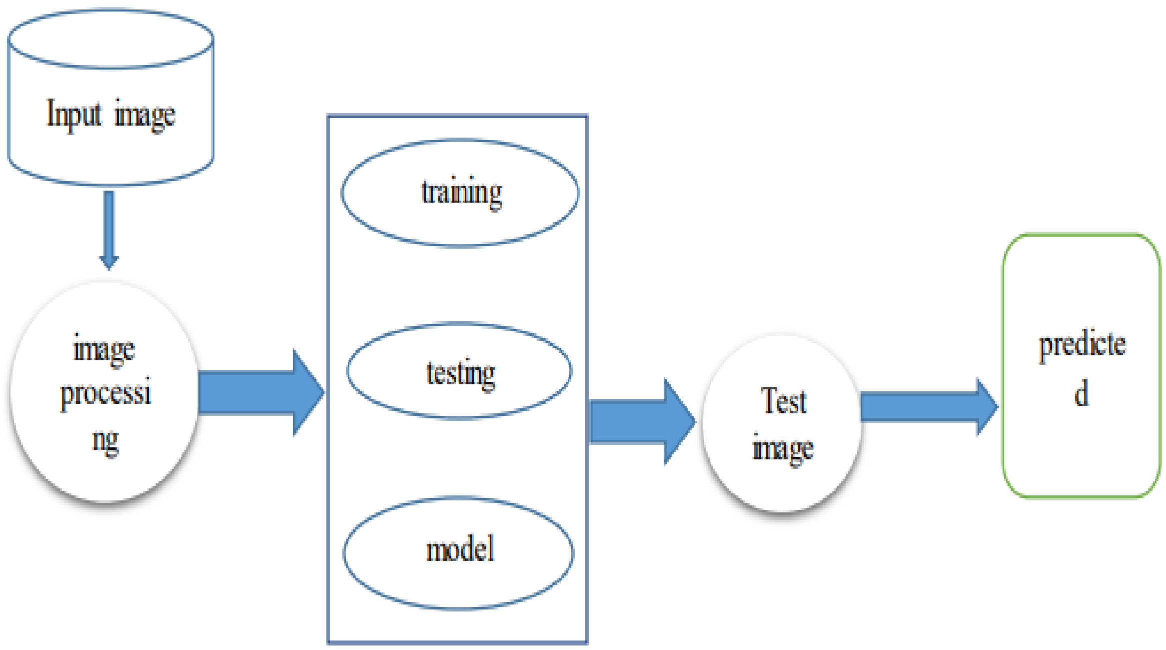
DATA FLOW DIAGRAM:

1. The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.
2. The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.
3. DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output. 4. DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction. DFD may be partitioned into levels that represent increasing information flow and functional detail.

Symbols	**Elements reference**
	Data Flow Process
	Process
	Source link
	Data Store


Preprocessing Data
Save the preprocessed file
Feeding the preprocessed data to convolutional neural network
Train
Saving the model
Prediction

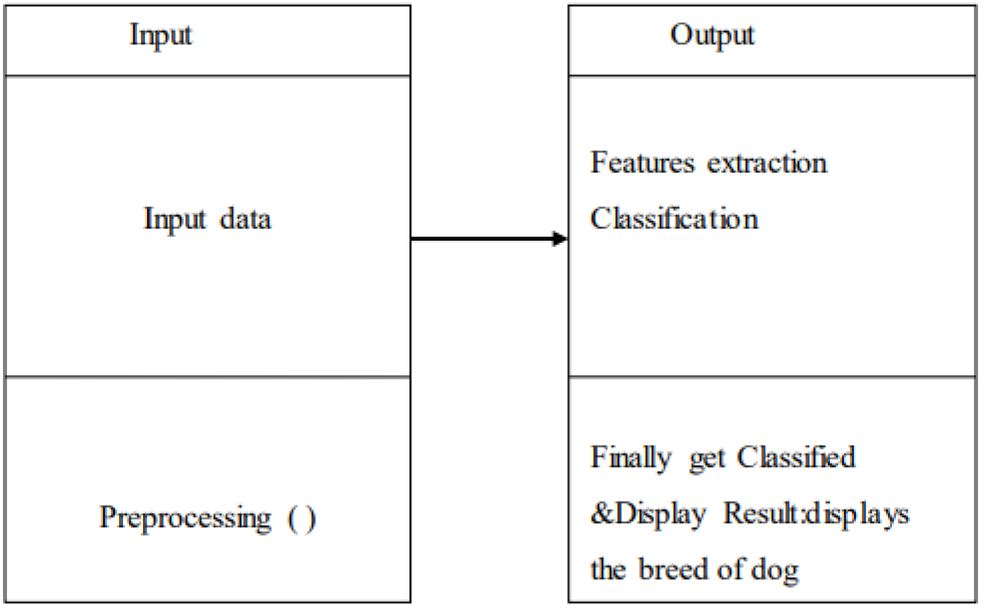
WORK FLOW DIAGRAMS



In this system architecture fig 5.7 ,customer to register/login to their website before uploading the image of the dog. After the completion of authentication process, user can upload the image that wanted to predicted.The image is processed. Then the model is trained and tested .The image is tested. Finally Melanoma is predicted.

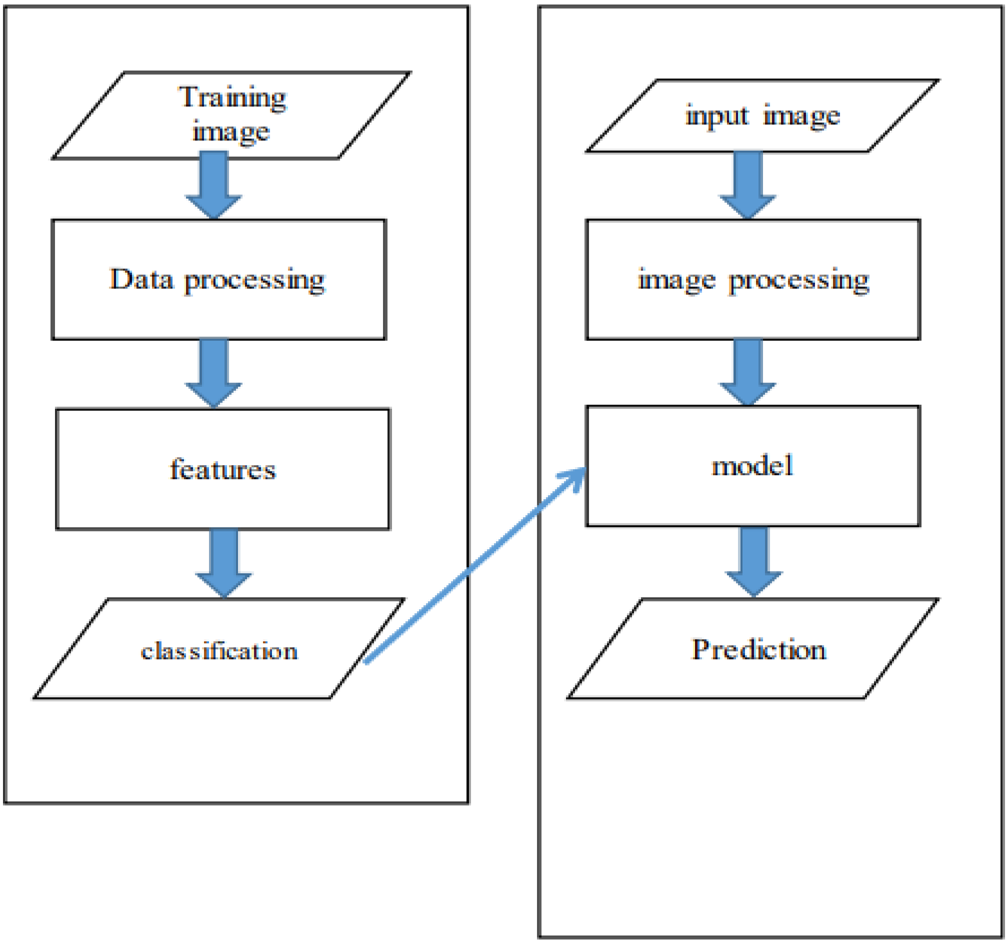
CLASS DIAGRAM

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information



SEQUENCE DIAGRAM

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.



Chapter 6

Conclusion and Future Scope

Convolutional neural networks (CNNs) have accomplished astonishing achievements across a variety of domains, including medical research, and an increasing interest has emerged in radiology. Although deep learning has become a dominant method in a variety of complex tasks such as image classification and object detection, it is not a panacea. Being familiar with key concepts and advantages of CNN as well as limitations of deep learning is essential in order to leverage it in radiology research with the goal of improving radiologist performance and, eventually, path.

In this project, it is seen that using deep learning there is no need of complex preprocessing techniques except for normalizing the pixel value, resizing and cropping. It bypasses all the preprocessing, segmentation and handcrafted feature ex-traction. It is observed that applying deep neural networks gives better result than conventional methods. For medical imaging,the most important challenge is that of acquiring labelled images. The ISBI challenge has provided with large number of datasets (ISIC Archive), which providesa common platform for researchers and academia to evaluate their work. Thefuture works might include: using a much larger dataset to reduce the risk of overfitting and performing additional regularization tweaks and fine-tuning of hyper-parameters. In many papers, it is shown that taking both global (deep)as well as local features into account during feature extraction provides betterresults. However, it is still an open issue to increase the accuracy rate. The goalshould be to target highest sensitivity while optimizing to increase specificity,thereby increasing the overall accuracy for practical applications

The salient features of this system are,

- User friendly
- Database Security
- Considerable reduction in the time consumption
- Security of records

BIBLIOGRAPHY

References

Following journals are referenced to create this project reports:

- *American Cancer Society. Cancer facts & figures 2016.* Atlanta, American Cancer Society 2016.

- *Giotis, N. Molders, S. Land, M. Biehl, M. F. Jonkman and N. Petkov*, "MED-NODE: A computer-assisted melanoma diagnosis system using non-dermoscopic images," Expert Systems with Applications, Elsevier, pp. 6578-6585, 2015.

- *Rogers, H. W. et al.* Incidence estimate of nonmelanoma skin cancer (keratinocyte carcinomas) in the US population, 2012.*JAMA Dermatology* **151.10**, 1081–1086 (2015)

- LeCun, Y., Bengio, Y. & Hinton, G. *Deep learning*. *Nature* **521**, 436–444 (2015)

- Ioffe, S. & Szegedy, C. *Batch normalization: accelerating deep network training by reducing internal covariate shift*. *Proc. 32nd Int. Conference on Machine Learning* 448–456 (2015)

- Rosado, B. *et al. Accuracy of computer diagnosis of melanoma: a quantitative meta-analysis*. *Arch. Dermatol.* **139**, 361–367, discussion 366 (2003)

- Gutman, D. *et al. Skin lesion analysis toward melanoma detection.International Symposium on Biomedical Imaging (ISBI)*, (International Skin Imaging Collaboration (ISIC), 2016)

- Abadi, M. *et al.* Tensorflow: large-scale machine learning on heterogeneous distributed systems. Preprint athttps://arxiv.org/abs/1603.04467 (2016)

- "*Melanoma Treatment–for health professionals (PDQ®)"*.National Cancer Institute. June 26, 2015. Archived from the original on 4 July 2015. Retrieved 30 June 2015

APPENDIX

Training

```
import matplotlib

matplotlib.use("Agg")

from keras.preprocessing.image import ImageDataGenerator

from keras.optimizers import Adam

from sklearn.model_selection import train_test_split

from keras.preprocessing.image import img_to_array

from keras.utils import to_categorical

from pyimagesearch.lenet import LeNet

from imutils import paths

import matplotlib.pyplot as plt

import numpy as np

import argparse

import random

import cv2

import os
```

```
ap = argparse.ArgumentParser()

ap.add_argument("-d", "--dataset", required=True,

help="path to input dataset")

ap.add_argument("-m", "--model", required=True,

help="path to output model")

ap.add_argument("-p", "--plot", type=str, default="plot.png",

help="path to output loss/accuracy plot")

args = vars(ap.parse_args())

EPOCHS = 25

INIT_LR = 1e-3

BS = 32

print("[INFO] loading images...")

data = []

labels = []



imagePaths = sorted(list(paths.list_images(args["dataset"])))

random.seed(42)

random.shuffle(imagePaths)
```

```
for imagePath in imagePaths:

image = cv2.imread(imagePath)

image = cv2.resize(image, (28, 28))

image = img_to_array(image)

data.append(image)

label = imagePath.split(os.path.sep)[-2]

label = 1 if label == "Melonoma" else 0

labels.append(label)

data = np.array(data, dtype="float") / 255.0

labels = np.array(labels)



 (trainX, testX, trainY, testY) = train_test_split(data,

labels, test_size=0.25, random_state=42)

trainY = to_categorical(trainY, num_classes=2)

testY = to_categorical(testY, num_classes=2)

aug = ImageDataGenerator(rotation_range=30, width_shift_range=0.1,

height_shift_range=0.1, shear_range=0.2, zoom_range=0.2,
```

```
horizontal_flip=True, fill_mode="nearest")

print("[INFO] compiling model...")

model = LeNet.build(width=28, height=28, depth=3, classes=2)

opt = Adam(lr=INIT_LR, decay=INIT_LR / EPOCHS)

model.compile(loss="binary_crossentropy", optimizer=opt,

metrics=["accuracy"])

print("[INFO] training network...")

H = model.fit_generator(aug.flow(trainX, trainY, batch_size=BS),

validation_data=(testX, testY), steps_per_epoch=len(trainX) // BS,

epochs=EPOCHS, verbose=1)

print("[INFO] serializing network...")

model.save(args["model"])

plt.style.use("ggplot")

plt.figure()

N = EPOCHS

plt.plot(np.arange(0, N), H.history["loss"], label="train_loss")

plt.plot(np.arange(0, N), H.history["val_loss"], label="val_loss")

plt.plot(np.arange(0, N), H.history["acc"], label="train_acc")
```

```
plt.plot(np.arange(0, N), H.history["val_acc"], label="val_acc")

plt.title("Training Loss and Accuracy on Melonoma /Not_Melonoma")

plt.xlabel("Epoch #")

plt.ylabel("Loss/Accuracy")

plt.legend(loc="lower left")

plt.savefig(args["plot"])
```

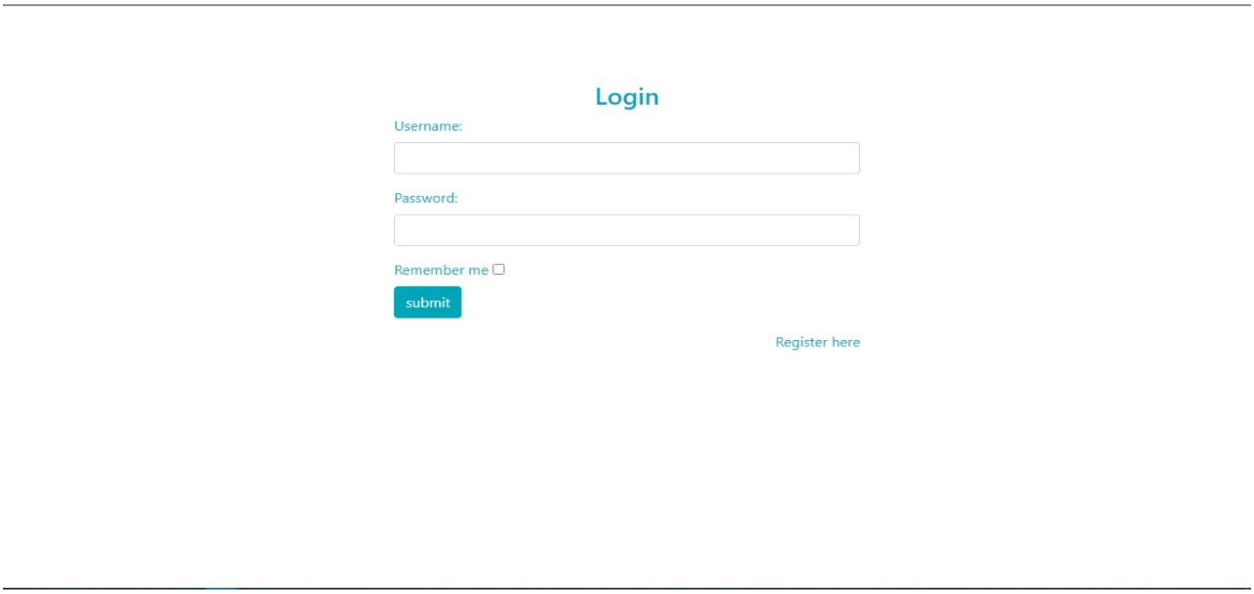
TESTING

```
from keras.preprocessing.image import img_to_array

from keras.models import load_model

import numpy as np

import argparse

import imutils

import cv2

ap = argparse.ArgumentParser()
```

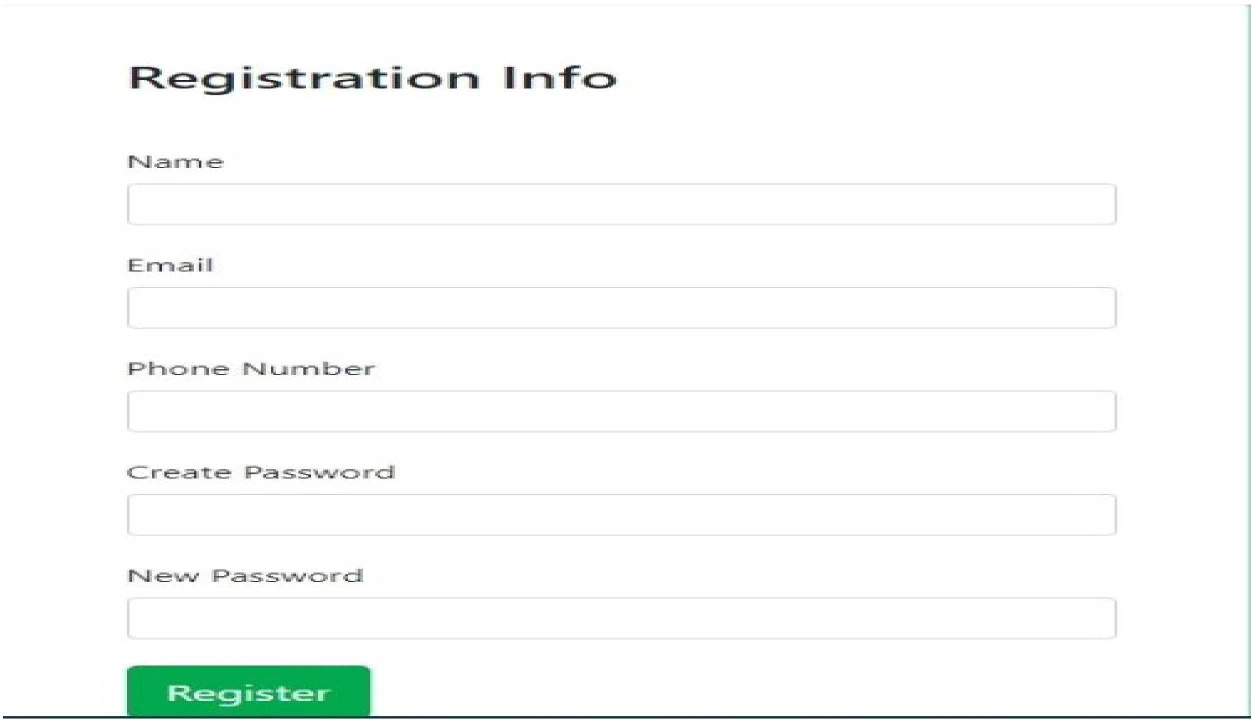
```
ap.add_argument("-m", "--model", required=True,

help="path to trained model model")

ap.add_argument("-i", "--image", required=True,

help="path to input image")

args = vars(ap.parse_args())

image = cv2.imread(args["image"])

orig = image.copy()

image = cv2.resize(image, (28, 28))

image = image.astype("float") / 255.0

image = img_to_array(image)

image = np.expand_dims(image, axis=0)

print("[INFO] loading network...")

model = load_model(args["model"])

 (Not_Melonoma, Melonoma) = model.predict(image)[0]

label = "Melonoma" if Melonoma > Not_Melonoma else "Not_Melonoma"

proba = Melonoma if Melonoma > Not_Melonoma else Not_Melonoma

label = "{}: {:.2f}%".format(label, proba * 100)

output = imutils.resize(orig, width=800)
```

```
cv2.putText(output, label, (10, 25),  cv2.FONT_HERSHEY_SIMPLEX,

0.7, (0, 255, 0), 2)

cv2.imshow("Output", output)

cv2.waitKey(0)
```

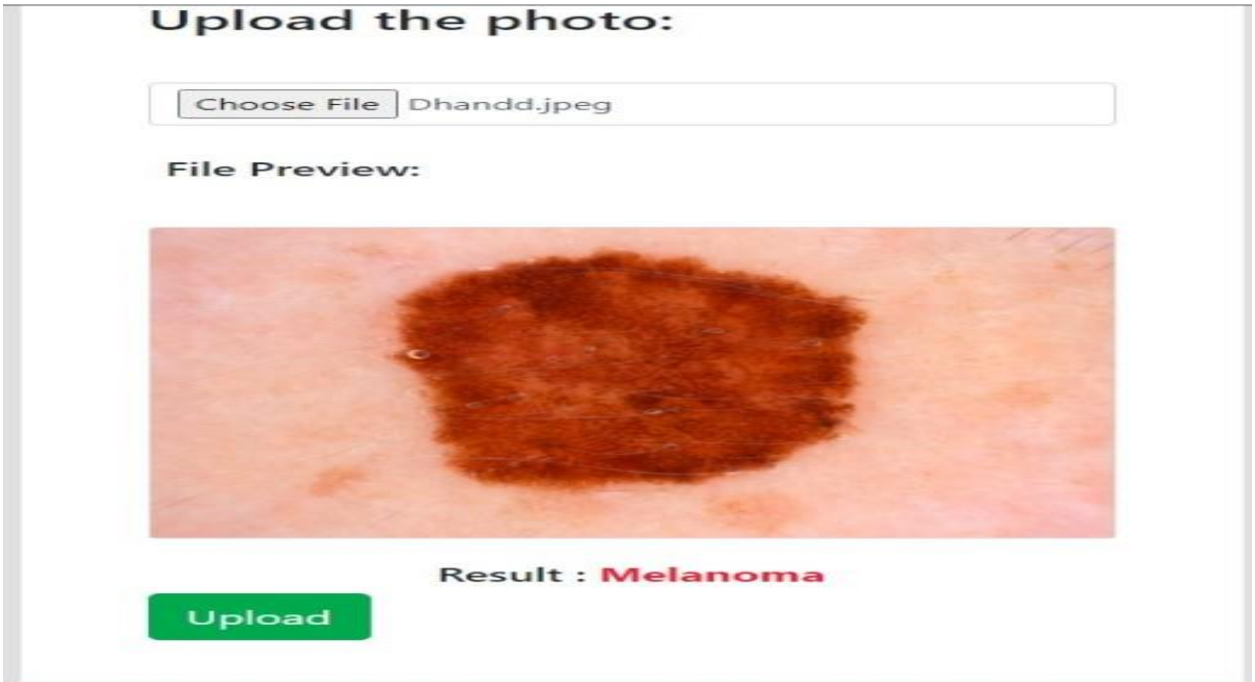
Output



(Fig: 8.3.1 Login)



(Fig: 8.3.2 Register)



(Fig: 8.3.3 Output)